



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN SISTEMAS

DESARROLLO DEL SISTEMA INFORMÁTICO PARA LA WEB
INTEGRANDO A JSP Y MARIADB COMO DBMS PARA LA
GESTIÓN DE FICHAS MÉDICAS DEL HOSPITAL BÁSICO “SAN
MARCOS” UTILIZANDO TECNOLOGÍA MÓVIL

TRABAJO DE TITULACIÓN: PROYECTO TÉCNICO

Para optar al Grado Académico de:

INGENIERO EN SISTEMAS INFORMÁTICOS

AUTOR: FAUSTO FABIÁN OROZCO CAJILEMA

TUTORA: ING. GERMANIA VELOZ

Riobamba-Ecuador

2017

©**2017**, Fausto Fabián Orozco Cajilema

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN SISTEMAS

El Tribunal del Trabajo de Titulación certifica que: La investigación: **DESARROLLO DEL SISTEMA INFORMÁTICO PARA LA WEB INTEGRANDO A JSP Y MARIADB COMO DBMS PARA LA GESTIÓN DE FICHAS MÉDICAS DEL HOSPITAL BÁSICO “SAN MARCOS” UTILIZANDO TECNOLOGÍA MÓVIL**, de responsabilidad del señor Fausto Fabián Orozco Cajilema, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Titulación, quedando autorizada su presentación.

NOMBRE	FIRMA	FECHA
Ing. Washington Luna DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA	_____	_____
Ing. M.Sc. Patricio Moreno DIRECTOR DE ESCUELA DE INGENIERÍA EN SISTEMAS	_____	_____
Ing. Germania Veloz DIRECTORA DEL TRABAJO DE TITULACIÓN	_____	_____
Ing. Fernando Mejía MIEMBRO DEL TRIBUNAL	_____	_____

Yo, Fausto Fabián Orozco Cajilema, soy responsable de las ideas, doctrinas y resultados expuestos en esta Investigación y el patrimonio intelectual del Trabajo de Titulación pertenece a la Escuela Superior Politécnica de Chimborazo.

Fausto Fabián Orozco Cajilema

DEDICATORIA

Escrito para la gloria de nuestro Amado Señor Jesucristo, quien es el dador de todo don perfecto. Es así como dedico este trabajo de titulación al Autor del Amor que es Amor, a Jesucristo Señor Todopoderoso, quien, por su gracia y por medio de la fe, recibí el don de la salvación y la vida eterna y que además es él que me cuida y sustenta para cumplir así con el propósito que Dios planificó en mi creación; a mi familia, principalmente a Hilda mi madre por su amor, entrega y dedicación por ser la guía en el sendero de cada acto que realizo; a mis hermanos Hugo y Paola, por su paciencia y por ser el incentivo para seguir adelante con este objetivo.

Fausto

AGRADECIMIENTO

Agradezco profundamente a Jesucristo, Amado Señor Mío, ya que por su gracia soy lo que soy; y su gracia no ha sido en vano para conmigo, antes he trabajado arduamente y mucho más; pero no yo, sino la gracia de Dios conmigo, y de esta manera cumplir con el plan y propósito de hacer su soberana voluntad buena, agradable y perfecta para mi vida.

A mi madre por su entrega, apoyo, comprensión, confianza y por sobre todas las cosas a su profundo amor, dando siempre lo mejor de su vida por buscar mi bienestar cada día.

A mi hermana Paola por su gran amor y apoyo incondicional, en cada decisión que tomo, y por estar a mi lado en cada momento hoy, mañana y siempre.

A mi hermano Hugo por estar siempre dispuesto a ayudarme y apoyarme en todas las metas que he tenido en mi diario caminar.

Mi agradecimiento por siempre a todas las personas que fueron designadas soberanamente por mi Amado Señor Jesucristo, quien es el dador de toda buena dadiva, para que estas vidas sirvan como un instrumento a través de quienes tu provisión llegó en cada momento de necesidad a mi vida.

A los hombres y mujeres que aman profundamente a Dios, amigas, amigos y a todas las personas que me motivaron con sus palabras de fe y esperanza para alcanzar un objetivo propuesto.

Fausto

TABLA DE CONTENIDO

ÍNDICE DE FIGURAS	ix
ÍNDICE DE GRÁFICOS	x
ÍNDICE DE TABLAS	xi
RESUMEN	xiii
ABSTRACT	xiv
INTRODUCCIÓN	1
CAPÍTULO I	8
1. MARCO TEÓRICO REFERENCIAL	8
1.1 Introducción.....	8
1.2 Dispositivos móviles.....	9
1.2.1 <i>Movilidad</i>	9
1.2.2 <i>Tamaño reducido</i>	9
1.2.3 <i>Comunicación inalámbrica</i>	10
1.2.4 <i>Interacción con las personas</i>	10
1.3 Tecnología de Desarrollo Web.....	10
1.3.1 <i>Lenguaje PHP</i>	11
1.3.2 <i>Lenguajes ASP y ASP.NET</i>	12
1.3.3 <i>JSP</i>	14
1.3.4 <i>Servidor Glassfish</i>	15
1.3.5 <i>Java EE7</i>	16
1.4 Tecnología de Desarrollo Móvil.....	18
1.4.1 <i>App nativas</i>	18
1.4.2 <i>Web App</i>	19
1.4.3 <i>Web App nativa</i>	20
1.5 Android.....	21
1.5.1 <i>AsyncTask en Android</i>	21
1.6 DBMS MariaDB	24
1.6.1 <i>Historia</i>	25
1.6.2 <i>Características</i>	25
1.6.3 <i>Tipos de servidores MariaDB</i>	26
1.6.4 <i>Instalación y puesta en marcha</i>	26

1.6.5	<i>El cliente de MariaDB en modo gráfico, creación de la base de datos</i>	27
1.6.6	<i>Conexión y uso de bases de datos MariaDB en lenguaje Java</i>	28
1.7	Ingeniería de Software	28
1.7.1	<i>Metodologías Tradicionales (no ágiles)</i>	29
1.7.2	<i>Metodologías Ágiles</i>	29
1.8	Metodología de desarrollo SCRUM	29
1.9	Terminología Médica	35
1.9.1	<i>Enfermedad</i>	35
1.9.2	<i>Codificación de enfermedades</i>	35
1.9.3	<i>Diagnóstico</i>	36
1.9.4	<i>Tratamiento</i>	38
1.9.5	<i>Historia Clínica</i>	39
1.9.6	<i>Nota de evolución o motivo de consulta</i>	39
	CAPÍTULO II	40
2.	MARCO METODOLÓGICO	40
2.1	Introducción	40
2.2	Planificación	40
2.2.1	<i>Usuarios del Sistema</i>	40
2.2.2	<i>Recopilación de los requerimientos para el desarrollo del Sistema</i>	41
2.2.3	<i>Historias de usuario</i>	42
2.2.4	<i>Iteraciones</i>	46
2.3	Diseño	55
2.3.1	<i>Arquitectura del sistema</i>	55
2.3.2	<i>Diagrama de componentes</i>	57
2.3.3	<i>Diagrama de despliegue</i>	57
2.3.4	<i>Esquema de la base de datos</i>	58
2.3.5	<i>Casos de Uso</i>	59
2.3.6	<i>Diagramas de Casos de Usos</i>	61
2.4	Codificación o Desarrollo	62
2.4.1	<i>Conexión de MariaDB con Android</i>	62
2.4.2	<i>Conexión de MariaDB con JSP</i>	64
2.5	Pruebas	65
2.5.1	<i>Pruebas de Aceptación</i>	65
	CAPÍTULO III	68

3.	MARCO DE RESULTADOS, DISCUSIÓN Y ANÁLISIS DE RESULTADOS	68
3.1	Estudio de aceptación del Sistema	68
3.2	Diseño de experimentos.....	68
3.3	Análisis estadístico en la inserción de datos	69
3.4	Análisis estadístico en la actualización de datos	78
3.5	Análisis estadístico en la consulta de datos	86
3.6	Análisis estadístico en la reserva de citas médicas.....	95
3.7	Tiempos totales	103
	CONCLUSIONES	106
	RECOMENDACIONES	107
	GLOSARIO	
	BIBLIOGRAFÍA	
	ANEXOS	

ÍNDICE DE FIGURAS

Figura 1-1. Desarrollo y programación de aplicaciones web.....	10
Figura 2-1. Arquitectura JSP.....	15
Figura 3-1. Contenedores de componentes.....	16
Figura 4-1. Desarrollo aplicaciones móviles.....	18
Figura 5-1. El ciclo de vida de la clase AsyncTask	22
Figura 6-1. Logo MariaDB	24
Figura 7-1. Creación de base de datos en MariaDB.....	28
Figura 8-1. Proceso de la Metodología SCRUM	30
Figura 9-1. Scrum Master	31
Figura 10-1. Product Owner.....	31
Figura 11-1. Scrum Team	32
Figura 12-1. Product Backlog	32
Figura 13-1. Sprint Planning Meeting.....	33
Figura 14-1. Sprint Planning Meeting.....	33
Figura 15-1. Sprint Process.....	34
Figura 16-1. Sprint Backlog.....	34
Figura 17-1. Sprint Retrospective	35
Figura 1-2. Capas del Sistema Gestión de Fichas Médicas	56
Figura 2-2. Diagrama de Componentes	57
Figura 3-2. Diagrama de Despliegue	58
Figura 4-2. Caso de Uso HU01.....	61
Figura 5-2. Caso de Uso HU02.....	62
Figura 6-2. Caso de Uso HU03.....	62

ÍNDICE DE GRÁFICOS

Gráfico 1-3 Tiempo de respuesta en la inserción de datos	70
Gráfico 2-3 Tiempo de respuesta medio en la inserción de datos	77
Gráfico 3-3 Tiempo de respuesta en la actualización de dato	78
Gráfico 4-3 Tiempo de respuesta medio en la actualización de datos	86
Gráfico 5-3 Tiempo de respuesta en la consulta de datos	87
Gráfico 6-3 Tiempo de respuesta medio en la consulta de datos	94
Gráfico 7-3 Tiempo de respuesta en la reserva de citas médicas	95
Gráfico 8-3 Tiempo de respuesta medio en la reservación de citas médicas	103
Gráfico 9-3 Tiempo de respuesta medio total	105

ÍNDICE DE TABLAS

Tabla 1-1: Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud.....	35
Tabla 1-2: Usuarios del Sistema Médico	40
Tabla 2-2: Product Backlog	44
Tabla 3-2: Sprint 1	46
Tabla 4-2: Sprint 2	48
Tabla 5-2: Sprint 3	50
Tabla 6-2: Sprint 4	52
Tabla 7-2: HU-01: Como Usuario del sistema pretendo autenticarme.	59
Tabla 8-2: HU-02: Como Administrador pretendo ingresar, modificar y eliminar a los usuarios del sistema.....	60
Tabla 9-2: HU-03: Como Administrador pretendo ingresar, modificar y eliminar a los días de la semana en los que atiende un doctor.....	61
Tabla 1-3: Tiempo de respuesta en la inserción de datos.....	69
Tabla 2-3: Prueba de normalidad de Shapiro-Wilk del tiempo de respuesta en la inserción de datos	71
Tabla 3-3: Decisión estadística de la normalidad de Shapiro-Wilk en la inserción de datos.....	72
Tabla 4-3: Prueba de Levene y Prueba t-Student en la inserción de datos	73
Tabla 5-3: Decisión estadística de la igualdad de varianzas en la inserción de datos.....	74
Tabla 6-3: Decisión estadística de la prueba t-Student para la igualdad de medias del tiempo en la inserción de datos	75
Tabla 7-3: Datos estadísticos t-Student en la inserción de datos	76
Tabla 8-3: Tiempo de respuesta en la actualización de datos	78
Tabla 9-3: Prueba de normalidad de Shapiro-Wilk del tiempo de respuesta en la actualización de datos	80
Tabla 10-3: Decisión estadística de la normalidad de Shapiro-Wilk en la actualización de datos	81
Tabla 11-3: Prueba de Levene y Prueba t-Student en la actualización de datos	82
Tabla 12-3: Decisión estadística de la igualdad de varianzas en la actualización de datos	83
Tabla 13-3: Decisión estadística de la prueba t-Student para la igualdad de medias del tiempo en la actualización de datos.....	84
Tabla 14-3: Datos estadísticos t-Student en la actualización de datos.....	84
Tabla 15-3: Tiempo de respuesta en la consulta de datos	86
Tabla 16-3: Prueba de normalidad de Shapiro-Wilk del tiempo de respuesta en la consulta de datos	88

Tabla 17-3: Decisión estadística de la normalidad de Shapiro-Wilk en la consulta de datos	89
Tabla 18-3: Prueba de Levene y Prueba t-Student en la consulta de datos.....	90
Tabla 19-3: Decisión estadística de la igualdad de varianzas en la consulta de datos	91
Tabla 20-3: Decisión estadística de la prueba t-Student para la igualdad de medias del tiempo en la consulta de datos	92
Tabla 21-3: Datos estadísticos t-Student en la consulta de datos.....	93
Tabla 22-3: Tiempo de respuesta en la reserva de citas médicas.....	95
Tabla 23-3: Prueba de normalidad de Shapiro-Wilk del tiempo de respuesta en la reservación de citas médicas	97
Tabla 24-3: Decisión estadística de la normalidad de Shapiro-Wilk en la reservación de citas médicas.....	98
Tabla 25-3: Prueba de Levene y Prueba t-Student en la reservación de citas médicas.....	99
Tabla 26-3: Decisión estadística de la igualdad de varianzas en la reservación de citas médicas	100
Tabla 27-3: Decisión estadística de la prueba t-Student para la igualdad de medias del tiempo en la reservación de citas médicas	101
Tabla 28-3: Datos estadísticos t-Student en la reserva de citas médicas	101
Tabla 29-3: Tiempo medio total.....	103

RESUMEN

El presente proyecto de investigación tuvo como objetivo desarrollar un Sistema Informático para la Gestión de Fichas Médicas del Hospital Básico San Marcos de la ciudad de Riobamba, para optimizar los tiempos de respuesta, consiguiendo la integración del sistema gestor de bases de datos (DBMS) MariaDB con la aplicación web de Páginas de Servidor de Java (JSP) y con la aplicación móvil Android. Para el análisis del Sistema Gestor de Base de Datos MariaDB y la tecnología JSP, se aplicaron aspectos y conceptos fundamentales de Base de Datos, JAVA en JSP y Android, mediante el entorno de desarrollo integrado (IDE) NETBEANS y el IDE Android Studio respectivamente. Se aplicó la metodología ágil SCRUM, por su fácil aplicación en la que se definieron los puntos más importantes conjuntamente con el cliente, como planificación, diseño, desarrollo, implementación y evaluación del sistema para asegurar la satisfacción del usuario. De los casos analizados se evidenció, con el uso del sistema, que fue mejorado el tiempo de respuesta medio total en un valor del 77,20 % en los procesos de gestión de las fichas médicas del hospital. Se concluye entonces desde el punto de vista funcional que el sistema es ágil y eficiente en los procesos de atención médica. Se sugiere implementar nuevos módulos de importancia, como hospitalización, emergencias, resultados de laboratorio y más, que permitan mejorar los servicios de la institución de salud.

PALABRAS CLAVES: <TECNOLOGÍA Y CIENCIAS DE LA INGENIERÍA>, <INGENIERÍA DE SOFTWARE>, <PÁGINAS DE SERVIDOR DE JAVA (JSP)>, <MARIADB (GESTOR DE BASE DE DATOS)>, <ENTORNO DESARROLLO INTEGRADO (IDE)>, <SCRUM (METODOLOGÍA DE DESARROLLO ÁGIL)>, <ANDROID (SISTEMA OPERATIVO MÓVIL)>, <NETBEANS (SOFTWARE)>.

ABSTRACT

The present research project aimed to develop a computer system for the management of medical records of the San Marcos Basic Hospital in the city of Riobamba, to optimize response times, achieving the integration of the database management system (DBMS) MariaDB with the web application of java server pages (JSP) and with the Android mobile application. For the analysis of the MariaDB database manager system and the JSP technology, fundamental aspects and concepts de database, JAVA in JSP and Android were applied through the NETBEANS integrated development environment (IDE) and the Android Studio respectively. The agile SCRUM methodology was applied due to its easy application in which the most important points were defined together with the client, such as planning, design, developmet, implementation and evaluation of the system to ensure user satisfaction. From the cases analyzed, it was evidenced, with the use of the system, that the total mean response time was improved in a value of 77,20 % in the management processes of the hospital's medical records. It is concluded from the functional point of view that the system is agile and efficient in the processes of medical care. It is suggested to implement new modules of importance, such as hospitalization, emergencies, laboratory results and more, to improve the services of the health institution.

KEY WORDS: TECHNOLOGY AND SCIENCES ENGINEERING, SOFTWARE ENGINEERING, JAVA SERVER PAGES (JSP), MARIADB (DATABASE MANAGER), INTEGRATED DEVELOPMENT ENVIRONMENT (IDE), SCRUM (AGIL DEVELOPMENT METHODOLOGY), ANDROID (MOBILE OPERATING SYSTEM), NETBEANS (SOFTWARE).

INTRODUCCIÓN

ANTECEDENTES

Las bases de datos en la actualidad son utilizadas para el almacenamiento estructurado de datos. Pues todo tipo de aplicaciones desde las más grandes, hasta los dispositivos móviles como son: celulares, SmartPhone, tablets o las agendas electrónicas utilizan tecnología de bases de datos para asegurar la integridad de los datos y facilitar la tarea tanto de los usuarios como de los programadores que las desarrollaron.

Una base de datos relacional es el modelo utilizado en la actualidad para modelar problemas reales y administrar datos. En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. La información puede ser recuperada o almacenada mediante “consultas” que ofrecen una amplia flexibilidad y poder para administrar la información (González, Á., 2011).

Los desarrolladores de aplicaciones cada vez adquieren más y nuevos conocimientos y es así que pueden dar solución a varios problemas surgidos en este ámbito, desde los años 70 hasta hoy en día, los Sistemas Gestores de Bases de Datos se han ido perfeccionando y ganando mucho terreno en el desarrollo de software, entre los cuales podemos incluir a los poderosos SQL Server, Oracle, PostgreSQL, MySQL, MariaDB, etc, independientemente de su licencia Libre o Pagada.

Como referencia de sistemas se puede citar al Sistema e-Salud ESPOCH que pertenece al Centro de Atención Integral en Salud de la Escuela Superior Politécnica de Chimborazo (CAISE), el mismo que brinda servicios médicos, como son los procesos relacionados a citas, fichas y recetas médicas para la comunidad politécnica (Tacuri Uquillas & Granizo Rodríguez, 2005).

Las ventajas que encontramos en la investigación de la tesis del Sistema e-Salud ESPOCH, permite que el sistema de Gestión de Fichas Médicas del Hospital Básico “San Marcos” se pueda desarrollar integrando a MariaDB como DBMS.

En la actualidad MariaDB es un reemplazo de MySQL con más funcionalidades y mejor rendimiento. MariaDB es un fork, es decir, es una ramificación de MySQL que nace bajo la licencia GPL. Esto se debe a que Oracle compró MySQL y cambió el tipo de licencia por un privativo, aunque mantuvieron MySQL Community Edition bajo licencia GPL. La compatibilidad de MariaDB con MySQL es prácticamente total y por si fuese poco tenemos mejoras de rendimiento y funcionalidad. MariaDB está diseñado para reemplazar a MySQL directamente ya que mantiene las mismas órdenes, APIs y bibliotecas (vozidea, 2013).

Hasta el día de hoy son varias las empresas de renombre mundial que están trabajando con MariaDB como servidor de Base de Datos, con cada uno de los proyectos que están ejecutando, entre las empresas más importantes encontramos a las siguientes: Google, Wikipedia, Redhat, Fedora, Hp, Tumblr, Opensuse. (Chávez & Alcantara, 2015).

El problema nace debido a que las fichas médicas son elaboradas manualmente, la asistente del doctor registra los signos vitales del paciente, crea o busca una historia clínica, hace la reservación de consultas y maneja la agenda del doctor. Todas estas tareas conllevan a que exista pérdida de tiempo, traspapelación de documentos o posibles errores al realizar manualmente cada uno de los procesos que están relacionados con la historia clínica.

La reservación de cita médica se realiza por medio de llamadas telefónicas o personalmente, las cuales se registran de forma manual en hojas, además los pacientes son citados en un mismo horario y las personas tienen que esperar hasta que les indiquen que pueden ingresar al consultorio.

A veces cuando se llega a la consulta no se encuentra el historial clínico del paciente y se pierde tiempo sabiendo que existen personas en la sala de espera por ser atendidos.

En cuanto al proceso de órdenes de exámenes médicos, los resultados de los mismos se extienden a los pacientes en papel, por este motivo se corre el riesgo de que una vez recibidos los resultados de los exámenes estos se pierdan o no se entreguen a tiempo. Además, actualmente no se puede evidenciar los resultados de los exámenes en línea lo cual genera una pérdida de tiempo y recursos para el médico tratante y el paciente.

La receta médica e indicaciones se lo realiza de forma manual, cuando se comete algún error dicha receta ya no sirve y hay que volver a escribirla.

El tiempo que lleva todas estas actividades es de aproximadamente 1 hora de acuerdo a la experiencia que los médicos del hospital han evidenciado al atender a los pacientes, motivo por el cual el Departamento de Administración (DA) del Hospital “San Marcos”, se ha visto en la imperiosa necesidad de implementar un sistema que agilite esta tarea, mismo que deberá ser de fácil manejo y amigable con el usuario, entregando así un producto software que cumpla con el objetivo previsto.

FORMULACIÓN DEL PROBLEMA

¿Cómo mejorar la gestión de Fichas Médicas del Hospital Básico "San Marcos" con el propósito de no perder tiempo al momento de atender a un paciente?

SISTEMATIZACIÓN DEL PROBLEMA

¿Cómo el médico registrará la consulta del paciente atendido?

¿Cómo se realizará una búsqueda minuciosa de la historia clínica del paciente en su libreta de registros?

¿De qué manera podrá el médico recetar sin pérdida de tiempo?

¿Cómo se verificará la confusión al momento de archivar los exámenes con los de otros pacientes?

¿De qué manera podrá llevar una historia clínica de un consultorio a otro con agilidad y en tiempo real?

¿Cómo el paciente podrá reservar su cita médica sin pérdida de tiempo y que pueda ser contestada a su teléfono vía mensaje?

JUSTIFICACIÓN DEL TRABAJO DE TITULACIÓN

JUSTIFICACIÓN TEÓRICA

Para el desarrollo del sistema de Gestión de Fichas Médicas del Hospital Básico “San Marcos” se ha considerado al DBMS MariaDB.

Un DBMS (Data Base Management System) Son las siglas en inglés para los Sistemas de Gestión de Bases de Datos (SGBD). Bajo este nombre se conoce a productos de fabricantes como Oracle, Sybase, Informix, Ingres, Borland, Microsoft, IBM, etc. Sistema de administración de bases de datos. Software que controla la organización, almacenamiento, recuperación, seguridad e integridad de los datos en una base de datos. Acepta solicitudes de la aplicación y ordena al sistema operativo transferir los datos apropiados (mastermagazine, 2010).

A continuación, se describen las ventajas que posee MariaDB.

Nuevos motores de almacenamiento, para la mayoría de usuarios lo interesante es Aria, que viene a reemplazar a MyISAM y también tenemos XtraDB que reemplaza a InnoDB. Los nuevos motores de almacenamiento son:

- Aria: Un motor de almacenamiento a prueba de fallos basado en MyISAM.
- XtraDB: El reemplazo del motor InnoDB basado en el plug-in de InnoDB.
- PBXT: Un motor de almacenamiento transaccional con una gran cantidad de nuevas y bonitas características.

- FederatedX: El reemplazo del motor Federated (vozidea, 2013).

Entre las ventajas más importantes ya descritas, es importante resaltar que el DBMS MariaDB permite almacenar grandes cantidades de datos, mejorando los tiempos de respuesta en la ejecución de nuestro sistema a desarrollar, logrando de ésta manera que el proceso de Gestión de Fichas Médicas se lo realice ágil y eficientemente.

Java es un lenguaje de programación y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems. Hay muchas aplicaciones y sitios web que no funcionarán a menos que tenga Java instalado y cada día se crean más. Java es rápido, seguro y fiable. Desde portátiles hasta centros de datos, desde consolas para juegos hasta súper computadoras, desde teléfonos móviles hasta Internet, Java está en todas partes (java, 2009).

El servidor de aplicaciones donde se desplegará el sistema es GlassFish éste es la implementación de referencia de Java EE y como tal soporta Enterprise JavaBeans, JPA, JavaServer Faces, JMS, RMI, Java Server Pages, servlets, además de otros. Todo lo anterior permite a los desarrolladores crear aplicaciones empresariales que son portables y escalables, además de su integración con las más antiguas tecnologías de este campo. GlassFish permite que componentes opcionales puedan ser instalados para servicios adicionales. Construido sobre un kernel modular y alimentado por OSGi, GlassFish se ejecuta en la parte superior de la implementación de Apache Felix (ecured, 2015).

La metodología a utilizar será SCRUM la misma que es un marco de trabajo de procesos que ha sido usado para gestionar el desarrollo de productos complejos desde principios de los años 90. Scrum no es un proceso o una técnica para construir productos; en lugar de eso, es un marco de trabajo dentro del cual se pueden emplear varias técnicas y procesos. Scrum muestra la eficacia relativa de las prácticas de gestión de producto y las prácticas de desarrollo, de modo que podamos mejorar (Schwaber & Sutherland, 2013).

JUSTIFICACIÓN APLICATIVA

El Hospital Básico “SAN MARCOS” como se mencionó anteriormente maneja una gran cantidad de datos de los pacientes, estos a su vez son administrados de forma manual, por lo que se requiere de un sistema con bases de datos que permita mejorar la administración y control de la información correspondiente a las fichas médicas de los pacientes.

El sistema de Gestión de Fichas Medicas estará desarrollado como aplicación web, haciendo uso del paradigma orientado a objetos, en el lenguaje de programación Java (versión EE 7),

requiriendo también del servidor GlassFish versión 4.1, usando la tecnología JSP y además del DBMS MariaDB, lo cual está explicado en el marco teórico.

Dicho sistema se encargará de almacenar en una base de datos toda la información de los pacientes que son atendidos en el Hospital; permitiendo así llevar un eficiente control de atenciones y ofreciendo el almacenamiento confiable de sus datos reduciendo de esta manera los tiempos en todos y cada uno de los procesos que conlleva la generación de las fichas médicas. Brindando de esta manera una mejor atención en dicha entidad de salud ya que permitirá optimizar el tiempo de atención a los pacientes por parte del personal encargado como son médicos, enfermeras, etc.

A fin de dar solución al problema planteado se considera implementar una aplicación web que podrá ser usado por un terminal de escritorio o móvil, es decir, la utilización de la tecnología móvil permitirá hacer la reservación, visualización y cancelación de citas médicas, como parte del proceso de gestión de las fichas médicas del Hospital Básico San Marcos de la Ciudad de Riobamba.

Considerando que actualmente está en auge dar una solución ágil, el sistema será desarrollado utilizando la metodología de desarrollo SCRUM, con procesos incrementales, basadas en iteraciones y revisiones continuas, siendo el objetivo principal el de elevar al máximo la productividad del equipo de desarrollo y que posee las siguientes etapas:

- Etapa de Análisis
- Etapa de Diseño
- Etapa de Desarrollo e Implementación
 - Módulo de Ingreso de Datos
 - Módulo de Actualización de Datos
 - Módulo de Eliminación de Datos
 - Módulo de Reportes de Datos
- Etapa de Desarrollo de la Metodología de Scrum
 - Proceso y Roles de Scrum
 - El proceso
 - Product Backlog
 - Sprint Planning
 - Sprint
 - Sprint Backlog
 - Daily Sprint Meeting
 - Demo y Retrospectiva

- Roles
 - Scrum Master
 - Product Owner (PO)
 - Team
- Etapa de Pruebas

Para el desarrollo modular se dividirá al sistema en 4 módulos denominados: Doctores, Pacientes, Enfermeras y Reportes.

Doctores: Serán los doctores quienes ingresarán los datos de enfermedades y subenfermedades, así como la nota de evolución, el diagnóstico y el tratamiento del paciente que está siendo atendido.

Pacientes: Se realizará el ingreso de los datos de los pacientes adjuntando una foto de los mismos, los datos de familiares o representantes, generación de fichas médicas.

Enfermeras: Las enfermeras podrán otorgar los turnos a cada paciente, registrar la toma de los signos vitales y demás requisitos necesarios para que pueda realizarse la atención por parte del doctor.

Reportes: Se mostrarán los listados de los pacientes a quienes se les ha creado su ficha médica, las recetas que se extiende al paciente, y principalmente la historia clínica del paciente.

Turnos: Los turnos serán otorgados de acuerdo a la disponibilidad de parte del médico y vía teléfono se enviará la respuesta con día y hora del mismo.

Los doctores o usuarios mediante la red y por medio de autenticación a nivel de la aplicación web tendrán acceso al servidor de aplicaciones y al servidor de base de datos formando la siguiente arquitectura del sistema.

OBJETIVOS

OBJETIVO GENERAL

Desarrollar un Sistema Informático para la Web integrando a JSP y MariaDB como DBMS para mejorar la agilidad y eficiencia de la Gestión de Fichas Médicas del Hospital Básico “San Marcos” utilizando tecnología móvil.

OBJETIVOS ESPECÍFICOS

- Analizar el DBMS MariaDB 10.0.20 para el desarrollo del sistema informático orientado a la Web.
- Desarrollar el Sistema para la Gestión de Fichas Médicas del Hospital Básico “San Marcos”, previo un análisis de la situación actual del hospital y su gestión de historias médicas y planeamiento de nuevos requerimientos del proceso.
- Implementar un módulo que permita otorgar turnos vía telefonía móvil con día y hora hacia el paciente.
- Realizar las pruebas necesarias con tiempos de respuesta óptimo en el adecuado desenvolvimiento del médico.

CAPÍTULO I

1. MARCO TEÓRICO REFERENCIAL

1.1 Introducción

Existen diversidad de DBMS (Data Base Management System), pero en esta ocasión se hablará acerca de MariaDB, así como también de la tecnología JSP (Java Server Pages) y la plataforma Android, de una manera sencilla pero clara.

Luego de haber realizado la respectiva investigación, se conoce que después de la adquisición de Sun Microsystems por Oracle, hubo un gran revuelo en la comunidad de desarrolladores MySQL y en el negocio de las bases de datos en general sobre el futuro de la comunidad de desarrollo de esa base de datos. (Dans, 2013)

De modo ingenioso, la comunidad MySQL fue capaz de crear una nueva base de datos partiendo de la rama pública y revivió el proyecto como un nuevo Sistema Gestor de Base de Datos (DBMS), que es prácticamente MySQL denominado ahora MariaDB. (Dans, 2013)

JSP es una tecnología java, empleada para crear páginas web dinámicas, que son ejecutadas en el servidor web, las páginas JSP contienen código HTML (Hyper Text Markup Language) o XML (eXtensible Markup Language), junto con etiquetas especiales determinadas. (Esteban, 2000).

Android es un SO (Sistema Operativo) orientado a dispositivos móviles desarrollado por Android Inc., compañía que fue adquirida por Google. Esta plataforma permite el desarrollo de aplicaciones de terceros a través del SDK (Software Development Kit), proporcionado por el mismo Google, y mediante el lenguaje de programación Java. Una alternativa es el uso del NDK (Native Development Kit) de Google para emplear el lenguaje de programación C. El código fuente de Android está disponible bajo diversas licencias de software libre y código abierto, destacándose la versión 2 de la licencia Apache.

1.2 Dispositivos móviles

El pensar en un dispositivo móvil como primera idea es un teléfono celular, la realidad es muy diferente porque hoy en día existen variedad de equipos móviles, así como PC portátiles, PocketPC, tabletas, etc.

Esta diversidad desencadena una problemática para quien debe programarlos, debido a las características específicas que poseen como una memoria determinada o un lenguaje que debe soportar.

Las características primordiales que tienen los dispositivos móviles se detallan a continuación:

- Aparatos pequeños.
- Se pueden transportar en el bolsillo del propietario o en un pequeño bolso.
- Capacidad de procesamiento.
- Conexión permanente o intermitente a una red.
- Memoria (RAM, tarjetas MicroSD, flash, etc.).
- Uso individual de una persona, tanto en posesión como en operación.
- Capacidad de interacción mediante la pantalla o el teclado. (Conocimientosweb.net, 2015)

Se puede definir con cuatro características básicas a los dispositivos móviles diferenciándolo de otros que, aunque parezcan similares, carecen de algunas de las características de los verdaderos dispositivos móviles. Estas cuatro características son:

1.2.1 Movilidad

Cualidad o característica básica que posee un dispositivo para ser transportado con frecuencia y facilidad. Los dispositivos móviles son adecuadamente pequeños como para ser trasladados y utilizados durante su transporte. (Prieto,et.al, 2011)

1.2.2 Tamaño reducido

Esta cualidad del dispositivo móvil permite la utilización fácil con una o dos manos sin necesidad de ninguna ayuda o soporte externo. A demás permite transportar el dispositivo cómodamente por parte de una persona. (Prieto,et.al, 2011)

1.2.3 Comunicación inalámbrica

Un concepto importante es el término inalámbrico, La comunicación inalámbrica se entiende por la capacidad que posee un dispositivo de enviar o recibir datos sin la necesidad de un cableado. (Prieto,et.al, 2011)

1.2.4 Interacción con las personas

La interacción es un proceso de uso que se establece entre el usuario con un dispositivo. El diseño de la interacción interviene disciplinas como la usabilidad y la ergonomía. (Prieto,et.al, 2011)

1.3 Tecnología de Desarrollo Web



Figura 1-1. Desarrollo y programación de aplicaciones web

Fuente: (h2tic, <https://www.h2tic.com/desarrollo-aplicaciones-web-malaga-jaen.html>)

Existen diferentes tecnologías para el desarrollo web a continuación se describen las más importantes:

- Diferentes tecnologías con un mismo propósito que es la de proporcionar contenido dinámico en la web.
- Todas se basan en:
 - Un *script* se ejecuta en el servidor web
 - Puede recibir información del usuario
 - Produce una página web dinámica
 - El *script* se encuentra intercalado con el código de una página HTML
- Emplean componentes o funciones para añadir más funcionalidades:
 - Acceso a bases de datos
 - Envío de correo electrónico

- Generación dinámica de imágenes
- Casi todas proporcionan un conjunto de objetos integrados accesibles desde los *scripts*

1.3.1 Lenguaje PHP

El lenguaje PHP (PHP Hypertext Pre-processor) es uno de los más antiguos (fue creado en 1995 por la empresa PHP Group) y utilizados en el diseño de páginas web que utilizan bases de datos. (Pérez, 2015)

Se trata de un lenguaje interpretado en el lado del servidor (no del navegador) por lo que permite la creación de páginas web dinámicas que pueden estar dentro de páginas en HTML. Como hemos indicado, no es un lenguaje compilado sino interpretado. (Pérez, 2015)

No todos los hostings o alojamientos web soportan PHP ya que requiere que el servidor tenga instalado IIS o Apache con las librerías de PHP. La extensión de los archivos de este lenguaje es la .php. Es uno de los lenguajes de programación web más populares por su rapidez y la facilidad de desarrollo. (Pérez, 2015)

Características

- Velocidad: PHP no solo es rápido al ser ejecutado, sino que no genera retrasos en la máquina, por esto no requiere grandes recursos del sistema. PHP se integra muy bien junto a otras aplicaciones, especialmente bajo ambientes Unix.
- Estabilidad: PHP utiliza su propio sistema de administración de recursos y posee de un sofisticado método de manejo de variables, conformando un sistema robusto y estable.
- Seguridad: PHP maneja distintos niveles de seguridad, estos pueden ser configurados desde el archivo .ini.
- Simplicidad: Usuarios con experiencia en C y C++ podrán utilizar PHP rápidamente. Además, PHP dispone de una amplia gama de librerías, y permite la posibilidad de agregarle extensiones. Esto le permite su aplicación en múltiples áreas, tales como encriptado, gráficos, XML y otras. (Red Gráfica Latinoamérica, 2010)

Ventajas

- Es un lenguaje sencillo y fácil de estudiar y aprender.
- Una de sus características es la rapidez.
- Lo soportan la mayoría de las plataformas de alojamiento web.
- Tiene ciertas características de los lenguajes orientados a objetos como la utilización de clases y herencias.
- Puede mezclarse con código HTML, aunque esto dificulta su lectura.
- Puede manejar ficheros y conectarse a distintas bases de datos (MySQL, Oracle, SQL Server, Informix, PostgreSQL, etcétera).
- Puede utilizarse para generar módulos binarios CGI.
- El software que permite soportarlo en los servidores de hosting es libre y gratuito.
- Está en continuo desarrollo y soporta numerosas funcionalidades.
- Existe numerosa documentación sobre el lenguaje en Internet por lo que es relativamente sencillo resolver los problemas que nos puedan surgir durante el desarrollo de un sitio web. (Pérez, 2015)

Desventajas

- Para poder ver y testear las páginas que vayamos creando es necesario disponer de un servidor web que soporte PHP.
- Al ser interpretado en el servidor, es más fácil que se colapse cuando el número de peticiones de descarga de páginas aumenta.
- Parte del contenido de las páginas puede no ser accesible a los navegadores, dificultando el posicionamiento de las páginas. (Pérez, 2015)

1.3.2 Lenguajes ASP y ASP.NET

La tecnología ASP surge en diciembre de 1996 con el lanzamiento del service pack 3 de Windows NT 4.0 como parte del IIS 3.0.

Este fue el primer acercamiento de Microsoft hacia la tecnología scripting en internet. La gran ventaja que tenía ASP era que uno podía diseñar su página y programar la lógica que haría que su página desplegara datos. ASP es una tecnología propietaria de Microsoft. (Pérez, 2015)

Características

- ASP no es en sí mismo un lenguaje de programación, sino más bien un marco sobre el que se construyen aplicaciones basadas en Internet.
- El tipo de servidores que emplean este, ASP 1.0 (distribuido con IIS lenguaje son aquellos que funcionan con sistema operativo de la familia de Windows NT. Afortunadamente, también podemos visualizar páginas ASP sobre Windows 95/98. Para escribir páginas ASP utilizamos un lenguaje de scripts, que se colocan en la misma página web junto con el código HTML.
- Comúnmente este lenguaje de scripts es Visual Basic Script, que deriva del conocido Visual Basic, aunque también se pueden escribir los scripts ASP en otro lenguaje: JScript, que deriva a su vez del conocido JavaScript.
- ASP ha pasado por cuatro interacciones mayores, ASP 1.0 (distribuido con IIS 3.0), ASP 2.0 (distribuido con IIS 4.0), ASP 3.0 (distribuido con IIS 5.0) y ASP.NET (parte de la plataforma .NET de Microsoft). Las versiones pre-.NET se denominan actualmente (desde 2002) como ASP clásico. (INFORMATICA, 2008)

Ventajas

- Se encarga de detectar el tipo de navegador utilizado por el cliente a la hora de realizar una petición al servidor y en consecuencia, determina la versión HTML que éste soporta.
- Es liviano.
- Se puede utilizar en cualquier computadora que esté conectada a la red que tenga instalado un navegador.
- Es muy fácil de programar y tiene muchas utilidades que con una breve línea de aprendizaje pueden ser modificadas a su gusto.
- Tiene la facilidad de conectarse con la base de datos, que hace que sea muy fácil.
- Permite a los proveedores de Web ofrecer aplicaciones de negocios interactivos y no simplemente meros contenidos publicables.
- Una de las limitaciones en el desarrollo con ASP es que con el tradicional utilizamos lenguajes de scripting no tipados como VBScript o JScript. Podemos instalar otros motores scripting que impongan verificación de tipos; sin embargo, no son universalmente conocidos o utilizados como los anteriores. (INFORMATICA, 2008)

Desventajas

- Una de las limitaciones en el desarrollo con ASP es que con el tradicional utilizamos lenguajes de scripting no tipados como VBScript o JScript. Podemos instalar otros motores scripting que impongan verificación de tipos; sin embargo, no son universalmente conocidos o utilizamos como los anteriores.
- Tiene que correr en PCs normales que tengan Windows y un servidor Web. (INFORMATICA, 2008)

1.3.3 JSP

JSP (Java Server Pages) es una tecnología que permite incluir código Java para crear páginas web dinámicas, brinda independencia de operar en diferentes plataformas y servidores de páginas Web, además combina el poder de la tecnología Java en el servidor con la facilidad de visualizar el contenido de las páginas HTML. (Sosa, 2014)

Características

- JSP amplía la tecnología de servlets para comprimir la programación requerida en el desarrollo de páginas Web dinámicas.
- JSP permite valorar mejor el estilo entre el contenido de la información y su presentación. (Sosa, 2014)

Ventajas

- Independencia del servidor Web.
- Substitución del API de los servlets por tags y scriptlets.
- Conseguir acceso al servidor Web en arquitecturas de partes múltiples (multi-tier).
- Separación del contenido estático del dinámico.
- Separación del contenido dinámico del formato de la presentación. (Sosa, 2014)

Elementos de JSP

Existen tres tipos de elementos JSP que se pueden insertar en una página web:

- **Código:** se puede "incrustar" código Java de distintos tipos (declaraciones de variables y/o métodos, expresiones, sentencias) para que el contenedor JSP lo ejecute.

- **Directivas:** controlar diferentes parámetros del servlet resultante de la traducción automática del JSP
- **Acciones:** Sirven para alterar el flujo normal de ejecución de la página, aunque tienen varios usos. (CCIA, 2005)

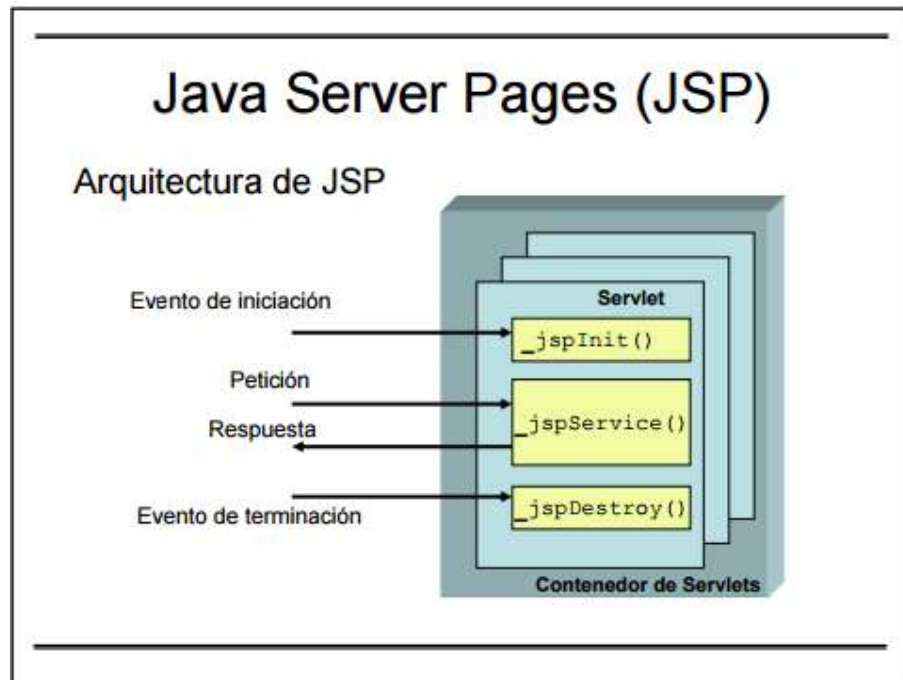


Figura 2-1. Arquitectura JSP
Fuente: (Sosa, 2014)

1.3.4 Servidor Glassfish

Servidor de aplicaciones de software libre compatible con Java EE permite ejecutar aplicaciones que siguen esta especificación, proporciona generalmente gran cantidad de funcionalidades built in de forma transparente al usuario de manera que no se necesite escribir código fuente.

Las funcionalidades son posibles debido a que los componentes se ejecutan en un espacio de ejecución virtual llamado dominio de ejecución. Como función principal es la de interponerse entre las llamadas que se hacen a los métodos de los beans y las implementaciones de los mismos. (Serra, 2010)

Modular, Integrable y Extendible

Según Serra (2010) Glassfish posee una arquitectura Modular, en la cual se puede descargar e instalar solamente los módulos necesarios para las apps, minimizando el tiempo de inicio, consumo de memoria y espacio en disco.

Basado en el modelo de componentes dinámico y completo para Java OSGi (Open Services Gateway Initiative), las aplicaciones y/o componentes de Glassfish pueden ser remotamente instalados, iniciados, actualizados, etc. sin necesidad de reiniciar el servidor. (Serra, 2010)

Dentro de una máquina virtual es posible ejecutar Glassfish sin necesidad de disponer un servidor de aplicaciones. Usar Glassfish como una librería es permitido más en la JVM, seleccionando solo lo necesario y probar en pequeñas aplicaciones webs sin necesidad de correr todo el AppServer, teniendo en cuenta las limitaciones de no tener el AppServer instalado. (Serra, 2010)

1.3.5 Java EE7

Un servidor de aplicaciones es un software de infraestructura que proporciona una serie de servicios a aplicaciones que corren en su interior.

Los Servlets son los programas Java que se ejecutan dentro de un servidor de aplicaciones Java EE, conocido como contenedor de Servlets. Cuando al contenedor le llega una petición http, comprueba si hay algún Servlet registrado para responder a dicha petición; en caso afirmativo, invocará dicho código.

Java EE – Contenedores de componentes

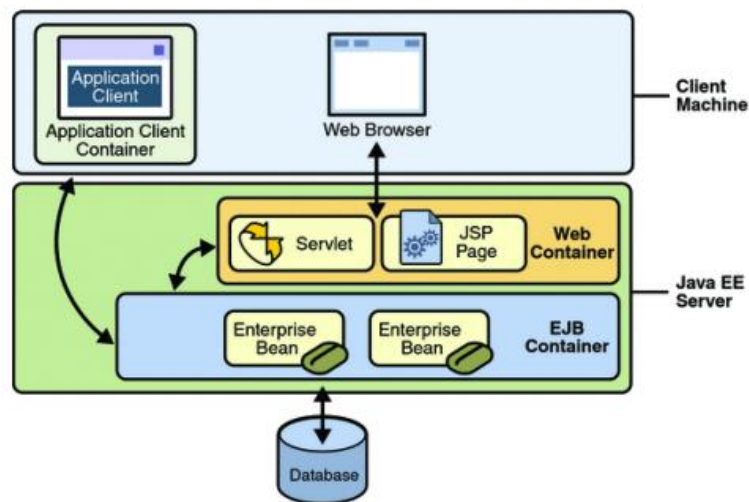


Figura 3-1. Contenedores de componentes

Fuente: The Java EE 6 Tutorial (2013).

Según Pavón (2013) a continuación se hace una comparación entre las características de los componentes y los objetos que presenta java al momento de implementar una solución óptima.

Componentes vs. objetos

Un componente se caracteriza por:

- *Ser una unidad de despliegue independiente*
 - *Encapsula sus características constituyentes respecto a su entorno*
 - *Las terceras partes no pueden acceder a los detalles de construcción del componente*
 - *No se implanta de manera parcial*
- *Ser una unidad de composición*
 - *Con componentes posiblemente desarrollados por otros*
 - *Debe ser suficientemente autocontenido*
 - *Especificaciones claras de lo que requiere y de lo que proporciona*
 - *Interacciona con su entorno a través de interfaces bien definidas*
- *No tener estado persistente*
 - *Un componente no se distingue de otras copias del mismo*
 - *Excepto atributos no funcionales como el número de serie*
 - *Por tanto, en un proceso se puede decir si hay o no un componente, pero no varias instancias del mismo*

Un objeto se caracteriza por:

- *Ser una unidad de instanciación; tienen una identidad única*
 - *No se instancia de manera parcial*
 - *La identidad es única y no cambia durante la vida del objeto*
- *Tener un estado*
 - *Se crea con un estado inicial que evoluciona durante la ejecución*
- *Encapsular su estado y comportamiento*
 - *Que está definido bien por una clase o por un objeto prototipo.*

1.4 Tecnología de Desarrollo Móvil



Figura 4-1. Desarrollo aplicaciones móviles

Fuente: (Web Studio Solutions, <https://webstudio.solutions/>)

En la actualidad el mercado de las aplicaciones móviles no para de crecer. Son muchas las empresas que no quieren dejar pasar la oportunidad de unirse a este negocio y **crear una aplicación** que dé respuesta a sus necesidades. Sin embargo, la mayoría desconocen qué tipos de aplicaciones móviles existen y cuál es la mejor para ellos. (LANCETALENT, 2014)

1.4.1 App nativas

Una **aplicación nativa** es la que se desarrolla de forma específica para un determinado sistema operativo, llamado *Software Development Kit* o SDK. Cada una de las plataformas, Android, iOS o Windows Phone, tienen un sistema diferente, por lo que si quieres que tu app esté disponible en todas las plataformas se deberán de crear varias apps con el lenguaje del sistema operativo seleccionado. (LANCETALENT, 2014)

Por ejemplo:

- Las apps para **iOS** se desarrollan con lenguaje Objective-C
- Las apps para **Android** se desarrollan con lenguaje Java
- Las apps en **Windows Phone** se desarrollan en .Net

Cuando hablamos de desarrollo móvil casi siempre nos estamos refiriendo a aplicaciones nativas. La principal ventaja con respecto a los otros dos tipos, es la posibilidad de acceder a todas las características del hardware del móvil: cámara, GPS, agenda, dispositivos de almacenamiento y otras muchas. Esto hace que la experiencia del usuario sea mucho más positiva que con otro tipo

de apps. Además, las aplicaciones nativas **no necesitan conexión a internet** para que funcionen. (LANCETALENT, 2014)

La **descarga e instalación** de estas apps se realiza siempre a través de las tiendas de aplicaciones (**app store de los fabricantes**). Esto facilita el proceso de marketing y promoción que explicaremos en próximos posts y que es vital para dar visibilidad a una app. Está claro que si el coste no es un obstáculo en tu empresa, o tienes la certeza de que tu app será rentable, la mejor opción será siempre el desarrollo de una aplicación nativa para cada plataforma (iOS, Android y Windows Phone). Si tu presupuesto es limitado, las aplicaciones web tienen también grandes ventajas para tu negocio. (LANCETALENT, 2014)

Ventajas

- Acceso completo al dispositivo
- Mejor experiencia del usuario
- Visibilidad en APP Store
- Envío de notificaciones o “avisos” a los usuarios
- La actualización de la app es constante

Desventajas

- Diferentes habilidades/idiomas/herramientas para cada plataforma de destino
- Tienden a ser más caras de desarrollar
- El código del cliente no es reutilizable entre las diferentes plataformas

1.4.2 Web App

Una **aplicación web** o **webapp** es la desarrollada con lenguajes muy conocidos por los programadores, como es el **HTML**, **JavaScript** y **CSS**. La principal ventaja con respecto a la nativa es la posibilidad de programar independiente del sistema operativo en el que se usará la aplicación. De esta forma se pueden ejecutar en diferentes dispositivos sin tener que crear varias aplicaciones. Las aplicaciones web **se ejecutan dentro del propio navegador web** del dispositivo a través de una URL. Por ejemplo, en Safari, si se trata de la plataforma iOS. El contenido se adapta a la pantalla adquiriendo un aspecto de navegación APP. (LANCETALENT, 2014)

¿Puede considerarse esto una APP? En realidad, la gran diferencia con una aplicación nativa (además de los inconvenientes que se muestran en la tabla) es que **no necesita instalación** por lo que no pueden estar visibles en app store y la promoción y comercialización debe realizarse de

forma independiente. De todas formas, se puede crear un acceso directo que sería como “instalar” la aplicación en el dispositivo. Las apps web móviles son siempre una buena opción si nuestro objetivo es **adaptar la web a formato móvil**. (LANCETALENT, 2014)

Ventajas

- El mismo código base reutilizable en múltiples plataformas
- Proceso de desarrollo más sencillo y económico
- No necesitan ninguna aprobación externa para publicarse (a diferencia de las nativas para estar visibles en app store)
- El usuario siempre dispone de la última versión
- Pueden reutilizarse sitios “responsive” ya diseñados

Desventajas

- Requiere de conexión a internet
- Acceso muy limitado a los elementos y características del hardware del dispositivo
- La experiencia del usuario (navegación, interacción, ...) y el tiempo de respuesta es menor que en una app nativa
- Requiere de mayor esfuerzo en promoción y visibilidad

1.4.3 Web App nativa

Una **aplicación híbrida** es una combinación de las dos anteriores, se podría decir que recoge lo mejor de cada una de ellas. Las apps híbridas se desarrollan con lenguajes propios de las webapp, es decir, HTML, JavaScript y CSS por lo que permite su uso en diferentes plataformas, pero también dan la posibilidad de acceder a gran parte de las características del hardware del dispositivo. La principal ventaja es que, a pesar de estar desarrollada con HTML, Java o CSS, **es posible agrupar los códigos y distribuirla en app store**. (LANCETALENT, 2014)

PhoneGap es uno de los frameworks más utilizados por los programadores para el desarrollo multiplataforma de aplicaciones híbridas. Otro ejemplo de herramienta para desarrollar apps híbridas es **Cordova**. (LANCETALENT, 2014)

Ventajas

- Es posible distribuirla en las tiendas de iOS y Android
- Instalación nativa pero construida con JavaScript, HTML y CSS
- El mismo código base para múltiples plataformas

- Acceso a parte del hardware del dispositivo

Desventajas

- Experiencia del usuario más propia de la aplicación web que de la app nativa
- Diseño visual no siempre relacionado con el sistema operativo en el que se muestre

1.5 Android

Android es un entorno operativo completo en capas que usa de base el kernel Linux e incluye vastas funciones, el objetivo de su implementación fue el área de dispositivos móviles, puede ser útil para otras plataformas y aplicaciones, la mayoría de las aplicaciones requieren guardar información, Android proporciona varias herramientas que pueden utilizarse para almacenar sus datos de forma local, debido a la necesidad de manejar una extensa cantidad de información se requiere un servidor externo para lo cual como gestor de datos de la aplicación se empleará MariaDB es un derivado de MySQL con licencia GPL. (Ableson, 2013)

1.5.1 AsyncTask en Android

La conexión desde un dispositivo con sistema Android con una base de datos MariaDB es perfectamente posible, se desarrollará un trabajo con una base de datos externa de acuerdo con las ventajas que posee Android cuenta con compilador Java admitiendo perfectamente trabajar con cualquier motor de base de datos que permita acceso mediante JDBC.

Al decir base de datos externa se refiere a que residirá un servidor externo con Windows o Linux como sistema operativo y el dispositivo accederá al servidor mediante la conexión de red o Internet.

Como es conocido Android obliga a hacer las operaciones de acceso a red en un hilo independiente, por tanto, se debe elegir una de las formas habituales de crear un nuevo hilo, por ejemplo, con un AsyncTask.

AsyncTask permite el uso adecuado y sencillo del subproceso de interfaz de usuario. Esta clase le permite realizar operaciones en segundo plano y publicar resultados en el subproceso de interfaz de usuario sin tener que manipular subprocesos y / o controladores. (Invarato, 2013)

Una tarea asincrónica se define por un cálculo que se ejecuta en un subproceso de fondo y cuyo resultado se publica en el subproceso de interfaz de usuario. Una tarea asincrónica está definida por 3 tipos genéricos, llamados Params, Progress y Result, y 4 pasos, llamados onPreExecute, doInBackground, onProgressUpdate y onPostExecute. (Developers, 2011)

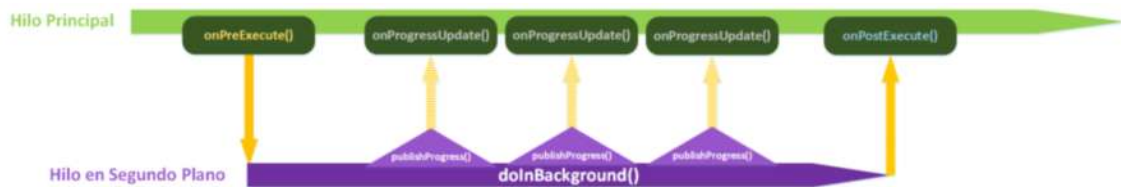


Figura 5-1. El ciclo de vida de la clase AsyncTask

Fuente: (Invarato, <https://jarroba.com/asynctask-en-android/>)

A continuación, se muestra una plantilla vacía de la clase AsyncTask en este código, el único método obligatorio es el doInBackground(), el resto se puede eliminar si no se va a usar.

```
new HiloEnSegundoPlano().execute("variable Tipo_empezarBackground", "variable
Tipo_empezarBackground", "...");
//Arrancamos el AsyncTask. el método "execute" envía datos directamente a doInBackground()
```

En esta línea lo que se está haciendo es ejecutar la clase asíncrona AsyncTask llamada HiloEnSegundoPlano.

```
private class HiloEnSegundoPlano extends AsyncTask <Tipo_empezarBackground,
                                                    Tipo_duranteBackground,
                                                    Tipo_terminarBackground> {

    public HiloEnSegundoPlano(){
        //TODO código del constructor
    }

    @Override
    protected void onPreExecute() {
        //TODO código del onPreExecute (Hilo Principal)
    }

    @Override
    protected Tipo_terminarBackground doInBackground(
        Tipo_empezarBackground... varEntrarBackground) {
        //TODO código del doInBackground (Hilo en Segundo Plano)

        //...

        publishProgress(valor_Tipo_duranteBackground);
        //Opcional. Para pasar el valor y llamar a onProgressUpdate()
```

```

        //...

        //...
        cancel(true); //Opcional. Para cancelar en vez de ejecutar al terminar
        onPostExecute(), ejecutar onCancelled().
        Comprobar si se ha cancelado con onCancelled()
        //...

        return valor_Tipo_terminarBackground;
    }

    @Override
    protected void onProgressUpdate(
        Tipo_duranteBackground... varDuranteBackground) {
        //TODO código del onProgressUpdate (Hilo Principal)
    }

    @Override
    protected void onPostExecute(Tipo_terminarBackground varTerminarBackground) {
        //TODO código del onPostExecute (Hilo Principal)
    }

    @Override
    protected void onCancelled (Tipo_terminarBackground varTerminarBackground) {
        //TODO código del onCancelled (Hilo Principal)
    }
} (Invarato, 2013)

```

Tipos genéricos de AsyncTask

Los tres tipos utilizados por una tarea asíncrona son los siguientes:

1. *Params*, el tipo de parámetros enviados a la tarea en ejecución.
2. *Progress*, el tipo de unidades de progreso publicadas durante la computación de fondo.
3. *Resultado*, el tipo de resultado del cálculo de fondo. (Developers, 2011)

No todos los tipos son siempre utilizados por una tarea asíncrona. Para marcar un tipo como no utilizado, simplemente utilice el tipo Void:

Los 4 pasos

Cuando se ejecuta una tarea asincrónica, la tarea pasa por 4 pasos:

1. `onPreExecute()`, invocado en el `onPreExecute()` de la interfaz de usuario antes de ejecutar la tarea. Este paso se utiliza normalmente para configurar la tarea, por ejemplo, mostrando una barra de progreso en la interfaz de usuario. (Developers, 2011)
2. `doInBackground(Params...)`, invocado en el subproceso de fondo inmediatamente después de que `onPreExecute()` termine de ejecutarse. Este paso se utiliza para realizar cálculos de fondo que pueden tardar mucho tiempo. Los parámetros de la tarea asincrónica se pasan a este paso. El resultado del cálculo debe ser devuelto por este paso y se pasará de nuevo al último paso. Este paso también puede usar `publishProgress(Progress...)` para publicar una o más unidades de progreso. Estos valores se publican en el subproceso de UI, en el `onProgressUpdate(Progress...)`. (Developers, 2011)
3. `onProgressUpdate(Progress...)`, invocado en el `onProgressUpdate(Progress...)` de la interfaz de usuario después de una llamada a `publishProgress(Progress...)`. La temporización de la ejecución no está definida. Este método se utiliza para mostrar cualquier forma de progreso en la interfaz de usuario mientras el cálculo de fondo sigue ejecutándose. Por ejemplo, se puede utilizar para animar una barra de progreso o mostrar registros en un campo de texto. (Developers, 2011)
4. `onPostExecute(Result)`, invocado en el subproceso de interfaz de usuario después de que finaliza el cálculo de fondo. El resultado del cálculo de fondo se pasa a este paso como un parámetro. (Developers, 2011)

1.6 DBMS MariaDB



Figura 6-1. Logo MariaDB

Fuente: (mariadb.org, <https://mariadb.org/about/logos/>)

MariaDB es una base de datos que tiene su origen en el código fuente de MySQL y está disponible bajo los términos de la licencia GPL v2, es decir que es de código abierto. Entonces debe quedar claro que la compatibilidad que existe entre MariaDB y MySQL es total, por ende, la migración de datos de la una base a la otra es fácil y no se necesita agregar ningún tipo de librerías para su correcto funcionamiento. Con respecto al rendimiento MariaDB presenta mejoras con relación a

MySQL, pues MariaDB afirma que tiene un optimizador de consultas mejorado, tal es el caso que algunas distribuciones Linux, han optado reemplazar a MySQL.

1.6.1 Historia

Según Dans (2013) MariaDB fue creada por Michael Monty Widenius quién se inspiró en el nombre de su hija menor para nombrar a esta base de datos que hoy por hoy ha superado al propio MySQL; además soporta todas las características de éste hasta la versión 5.5, de ahí en adelante inicia su desarrollo independiente con la versión 10, mejorando las características que se han agregado a MySQL desde que Oracle es su propietario. Poco a poco MariaDB ha ido posicionándose en el mercado del software, reuniendo a una gran comunidad de desarrolladores en todo el mundo.

1.6.2 Características

- Adicionalmente a los motores estandar MyISAM, Blackhole, CSV, Memory, y Archive, los siguientes motores están incluidos en los paquetes binarios y fuente de MariaDB:
 - Aria
 - XtraDB (reemplazo para InnoDB)
 - PBXT
 - FederatedX (reemplazo para Federated)
 - OQGRAPH — *nuevo en 5.2*
 - SphinxSE — *nuevo 5.2*
 - IBMDB2I. Removido por Oracle en MySQL 5.1.55 pero nosotros mantenemos el código en MariaDB.
 - TokuDB
 - Cassandra (MariaDB 10.0)
 - CONNECT (MariaDB 10.0)
 - SEQUENCE (MariaDB 10.0)
 - Spider (MariaDB 10.0) (MARIADB, 2011)

- Mejoras de velocidad sobre todo en consultas complejas cuando se usa el motor de almacenamiento Aria, ya que Aria cachea los datos de tablas temporales en memoria, lo que supone un rendimiento frente al uso del disco duro (que es lo que emplea MyISAM). (vozidea, 2013)
- Se añaden nuevas tablas de sistema (`INFORMATION_SCHEMA`) para almacenar estadísticas que nos pueden ayudar a optimizar las bases de datos. (vozidea, 2013)
- El sistema para manejar las conexiones se ha mejorado, ya que implementa el sistema pool-of-threads de MySQL 6.0 con el que podemos tener más de 200.000 conexiones a MariaDB. (vozidea, 2013)
- Contiene más pruebas en la suite de pruebas y sus parches. (vozidea, 2013)

1.6.3 Tipos de servidores MariaDB

WPN-XM SERVER

Es un servidor de código abierto el cual reemplaza a MySQL por MariaDB, además incorpora el debugger **XDebug** y el profiler **XHprof**, por lo que es muy apropiado para tareas de programación de alto rendimiento.

La base de la pila de servidores de WPN-XM Server consta de: Nginx, PHP y MariaDB. XDebug y Webgrind se añadieron a efectos de analizar y depurar; phpMyAdmin y Adminer para fines de administración de bases de datos MySQL; memcached y UCPA para almacenar en caché; ZeroMQ para la magia zócalo. WPI-XM proporciona una interfaz web para la administración del servidor y un panel de control del servidor (uso de la bandeja) que hace que arranque y parada de servidores fáciles.

La pila tiene un "Componente de registro de software", que es una base de datos en línea de todos los componentes de software proporcionados por el proyecto. Este registro se mantiene a través del "Registro de actualización", que permite el rastreo de los sitios de los proveedores para las versiones más recientes de su software y actualiza el registro en consecuencia. (GITHUB, 2015)

1.6.4 Instalación y puesta en marcha

MariaDB es un motor de base de datos que puede ser instalado en cualquiera de las plataformas, Windows o Linux.

La instalación de MariaDB es sencilla, se pueden dejar los valores por defecto que nos brinda el mismo proceso de instalación o se puede personalizarlo como en este caso.

Los pasos para la instalación son:

1. En la primera pantalla hacer clic en siguiente como se muestra en la Figura 18-1 (ver anexo A).
2. Se deberá aceptar las políticas de uso y licencia, después hacer clic en siguiente ver figura 19-1 (ver anexo A).
3. En este paso de la instalación no se cambia nada y se presione siguiente ver figura 20-1 (ver anexo A).
4. En este momento se debe escribir una contraseña para el administrador de la base de datos, además para este caso es muy importante habilitar la opción de acceso remoto desde cualquier cliente, debido a que se implementará una aplicación para dispositivos móviles, la misma que debe tener acceso a la base de datos creada en este motor. A continuación, dar clic en siguiente ver figura 21-1 (ver anexo A).
5. En este instante escribir el nombre del servidor y el puerto por el cual escuchará el DBMS, una vez realizado esto hacer clic en siguiente ver figura 22-1 (ver anexo A).
6. En este paso no hacer absolutamente nada, solo dar clic en siguiente ver figura 23-1 (ver anexo A).
7. Finalmente hacer clic en Instalar para que inicie la instalación del motor de base de datos MariaDB. ver figura 24-1 (ver anexo A).
8. Se visualizará una pantalla que indica que la instalación ha finalizado satisfactoriamente. ver figura 25-1 (ver anexo A).

1.6.5 El cliente de MariaDB en modo gráfico, creación de la base de datos

La creación de la base de datos en MariaDB es muy similar a la de MySQL, después de ejecutar la aplicación y cuando es por primera vez, se debe configurar el servidor para establecer la conexión entre la base de datos y el servidor, como se muestra en la Figura 26-1 (ver anexo B); nombre del host, usuario y contraseña y finalmente el puerto por el cual se tiene acceso al servidor.

La interfaz de HeidiSQL, se encuentra dividida en dos secciones, para crear la base de datos se debe hacer clic derecho sobre el lado izquierdo de la pantalla y elegir la opción crear nuevo y en el siguiente menú desplegable, base de datos; entonces aparecerá una ventana donde se pone el nombre a la base de datos a crear y ya está lista la base de datos para iniciar el trabajo. Ver Figura 27-1 (ver anexo B)

1.6.6 Conexión y uso de bases de datos MariaDB en lenguaje Java

“El acceso a bases de datos desde Java se realiza mediante el estándar JDBC (Java Data Base Connectivity), que permite un acceso uniforme a las bases de datos independientemente del SGBD. De esta manera, las aplicaciones escritas en Java no necesitan conocer las especificaciones de un SGBD en particular, basta con comprender el funcionamiento de JDBC. Cada SGBD que quiera utilizarse con JDBC debe contar con un adaptador o controlador”. (Dataprix, 2006)

En el desarrollo de la aplicación web y la aplicación móvil se utiliza el driver JDBC de Oracle para establecer la conexión con la base de datos en MariaDB.

En la Figura 7-1 se muestra la estructura del funcionamiento del JDBC para poder entenderlo de una manera simple.

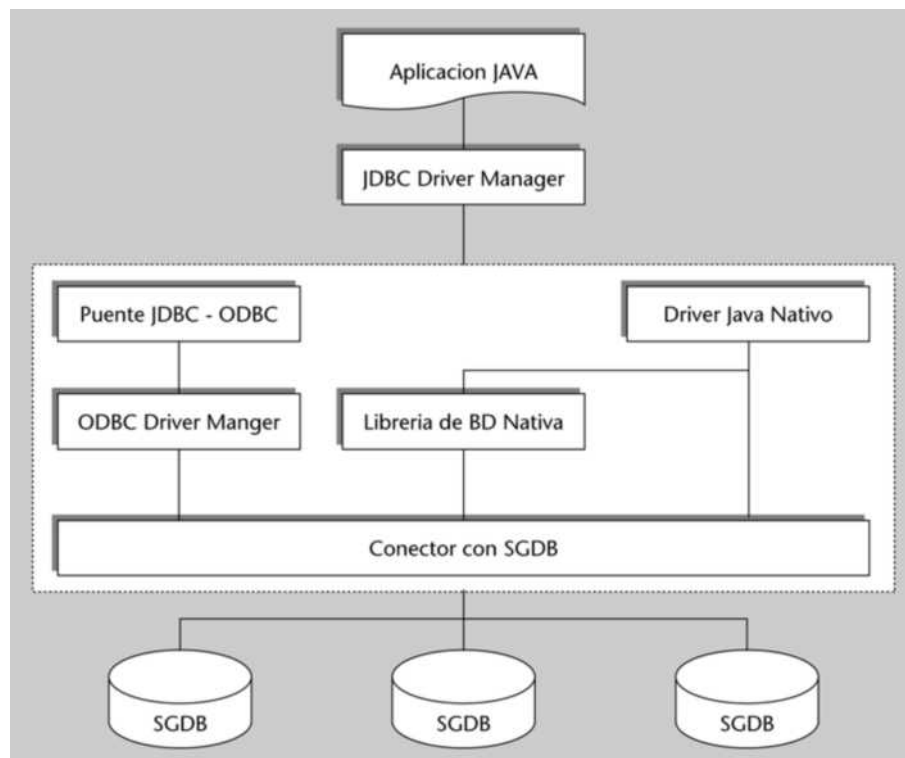


Figura 7-1. Creación de base de datos en MariaDB

Fuente: (Dataprix, <http://www.dataprix.com/2-conexion-uso-bases-datos-lenguaje-java>)

1.7 Ingeniería de Software

Ingeniería del Software es la aplicación *práctica* del conocimiento científico en el diseño y construcción de programas de computadora y la *documentación* asociada requerida para desarrollar, operar (funcionar) y mantenerlos. Se conoce también como desarrollo de software o producción de software. (Pressman, 2008)

1.7.1 Metodologías Tradicionales (no ágiles)

Las metodologías no ágiles son aquellas que están guiadas por una fuerte planificación durante todo el proceso de desarrollo; llamadas también metodologías tradicionales o clásicas, donde se realiza una intensa etapa de análisis y diseño antes de la construcción del sistema. (Universidad Politécnica de Valencia, 2010)

Entre las metodologías tradicionales más destacadas se pueden nombrar:

- Genérico
- Cascada - Secuencial/ Water fall
- Construcción de Prototipo
- Espiral
- Incremental

1.7.2 Metodologías Ágiles

Un proceso es ágil cuando el desarrollo de software es **incremental** (entregas pequeñas de software, con ciclos rápidos), **cooperativo** (cliente y desarrolladores trabajan juntos constantemente con una cercana comunicación), **sencillo** (el método en sí mismo es fácil de aprender y modificar, bien documentado), y **adaptable** (permite realizar cambios de último momento). (Universidad Politécnica de Valencia, 2010)

Entre las metodologías ágiles más importantes se encuentran las siguientes:

- Extreme Programming
- Scrum
- Familia de Metodologías Crystal
- Feature Driven Development
- Proceso Unificado Rational, una configuración ágil
- Dynamic Systems Development Method
- Adaptive Software Development
- Open Source Software Development

1.8 Metodología de desarrollo SCRUM

“Scrum es una metodología ágil de gestión de proyectos de desarrollo de software, basada en un proceso de trabajo constante, iterativo e incremental.” (Bahit, E, 2011)

Scrum consiste en aplicar de manera constante una secuencia de pasos, en conjunto con el equipo de trabajo para conseguir los mejores resultados del proyecto en el cual se está trabajando. Las entregas del producto son parciales de acuerdo a la priorización que le da el usuario, según sean sus necesidades.

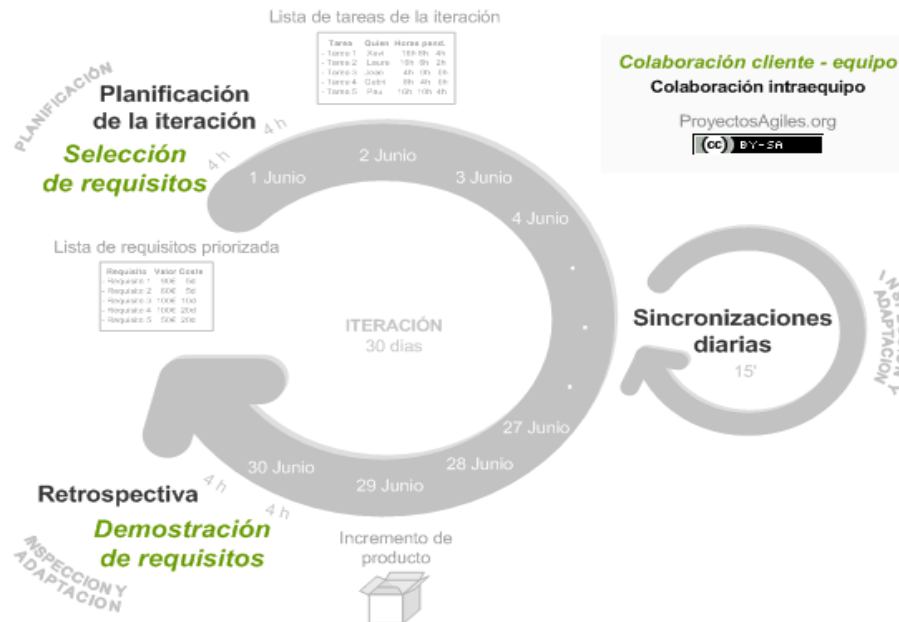


Figura 8-1. Proceso de la Metodología SCRUM

Fuente: (ProyectosAgiles.org, <https://proyectosagiles.org/que-es-scrum/>)

Roles de Scrum

La metodología Scrum, se preocupa por obtener un software de calidad, para lo cual define claramente las características con las que debe cumplir el producto final; para conseguir este objetivo está compuesto de los siguientes roles:

Scrum Master. - Es la persona que guía al equipo de trabajo y controla que se cumplan todas las tareas que se han propuesto, además se encarga de minimizar los problemas que se presentan durante el desarrollo del software. Realiza un trabajo conjunto con el Product Owner. (Softeng, 2010)

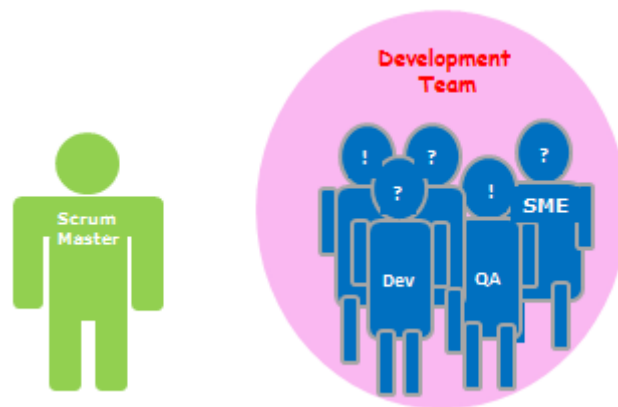


Figura 9-1. Scrum Master

Fuente: (softwaretestingmaterial.com, <http://www.softwaretestingmaterial.com/agile-scrum-methodology/>)

Product Owner. - Está en representación del usuario que utilizará el software y es responsable de que el equipo conozca las funcionalidades con las que debe contar el sistema.

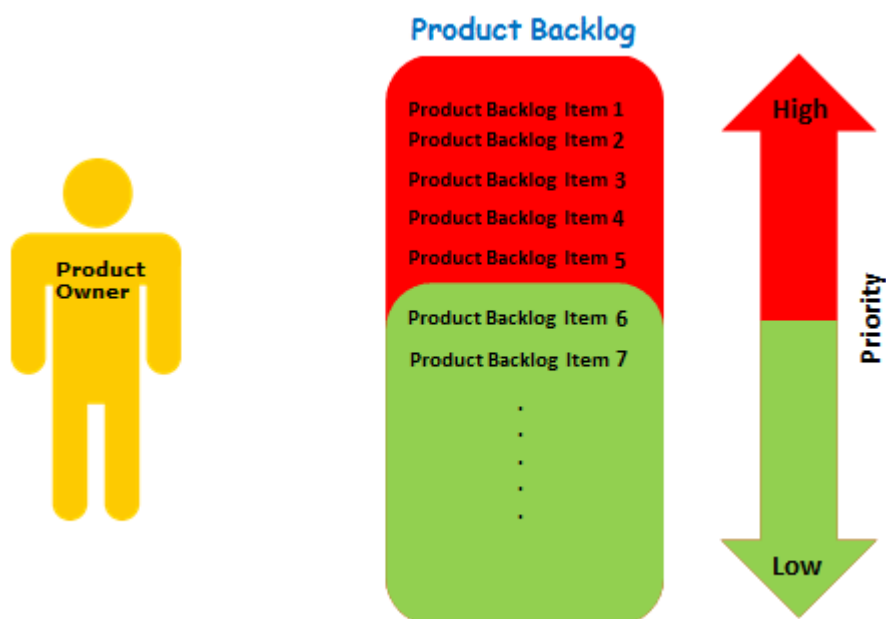


Figura 10-1. Product Owner

Fuente: (softwaretestingmaterial.com, <http://www.softwaretestingmaterial.com/agile-scrum-methodology/>)

Scrum Team. - Es todo el equipo de profesionales que se encarga del desarrollo del software.

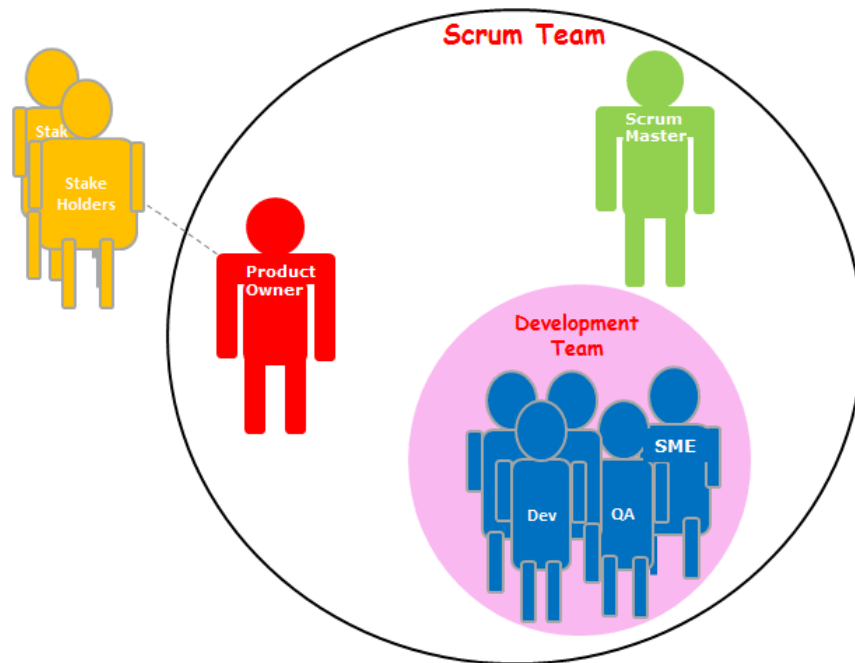


Figura 11-1. Scrum Team

Fuente: (softwaretestingmaterial.com, <http://www.softwaretestingmaterial.com/agile-scrum-methodology/>)

El proceso de Scrum

Planificación de la Iteración En esta etapa se realiza el product backlog y el sprint planning.

Product Backlog Es la lista priorizada de las historias de usuario.

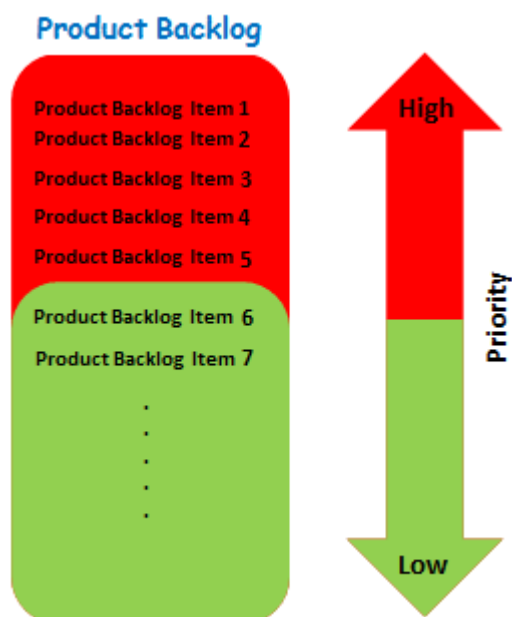


Figura 12-1. Product Backlog

Fuente: (softwaretestingmaterial.com, <http://www.softwaretestingmaterial.com/agile-scrum-methodology/>)

Sprint Planning Es la reunión donde el equipo selecciona libremente las historias de usuario que considera que las puede desarrollar.

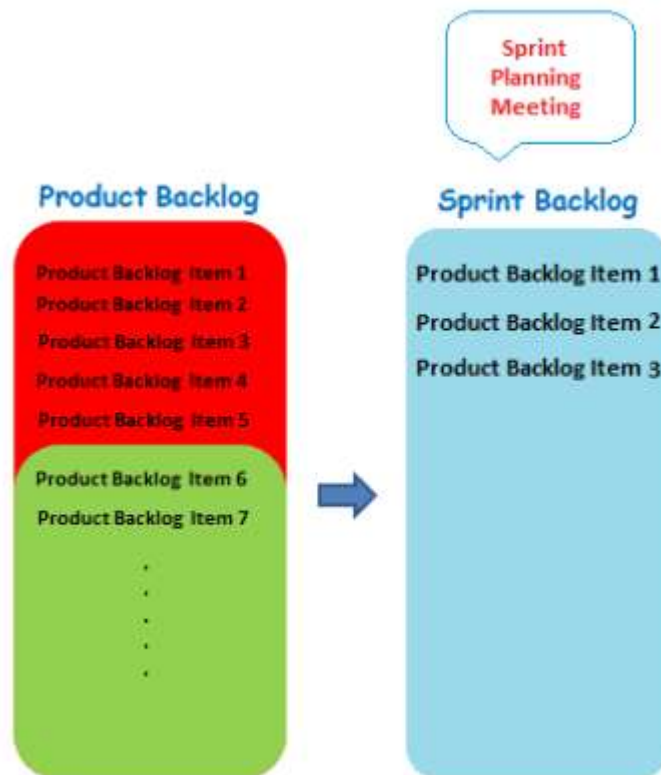


Figura 13-1. Sprint Planning Meeting

Fuente: (softwaretestingmaterial.com, <http://www.softwaretestingmaterial.com/agile-scrum-methodology/>)

Ejecución de la Iteración En esta etapa se realiza el sprint y el sprint backlog.

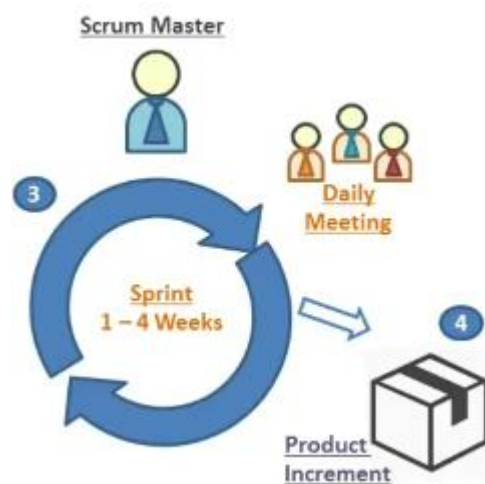


Figura 14-1. Sprint Planning Meeting

Fuente: (Join Academia, <https://www.youtube.com/watch?v=edGD-pbE1Mo>)

Sprint Es la iteración que se ha señalado para ejecutar el desarrollo de las historias de usuario, es decir para llevarlas al nivel operativo. (Softeng, 2010).

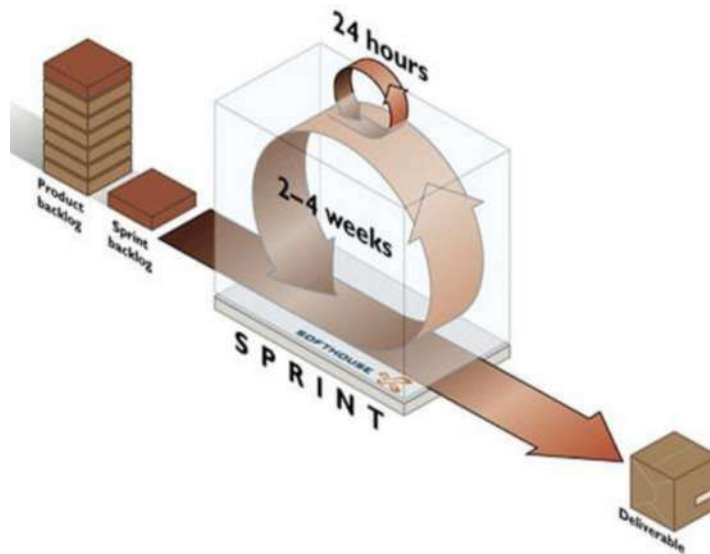


Figura 15-1. Sprint Process

Fuente: (codeproject.com, <https://www.codeproject.com/Articles/551161/Agile-Testing-Scrum-and-eXtreme-Programming>)

Sprint Backlog Lista de tareas de cada Sprint.

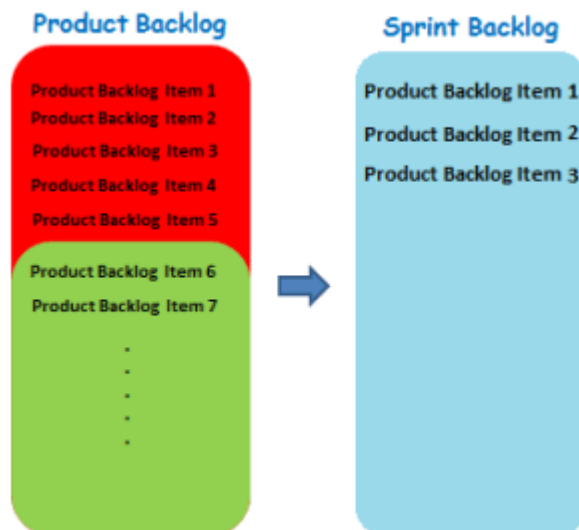


Figura 16-1. Sprint Backlog

Fuente: (softwaretestingmaterial.com, <http://www.softwaretestingmaterial.com/agile-scrum-methodology/>)

Inspección y Adaptación En este punto se realiza la demostración y retrospectiva

Demostración Es la reunión que se debe realizar al final de cada sprint para que cada miembro del equipo presente sus actividades realizadas.

Retrospectiva En esta segunda parte, el equipo hace un análisis sobre las cosas que se han hecho bien y los procesos que se pueden mejorar. (Softeng, 2010).

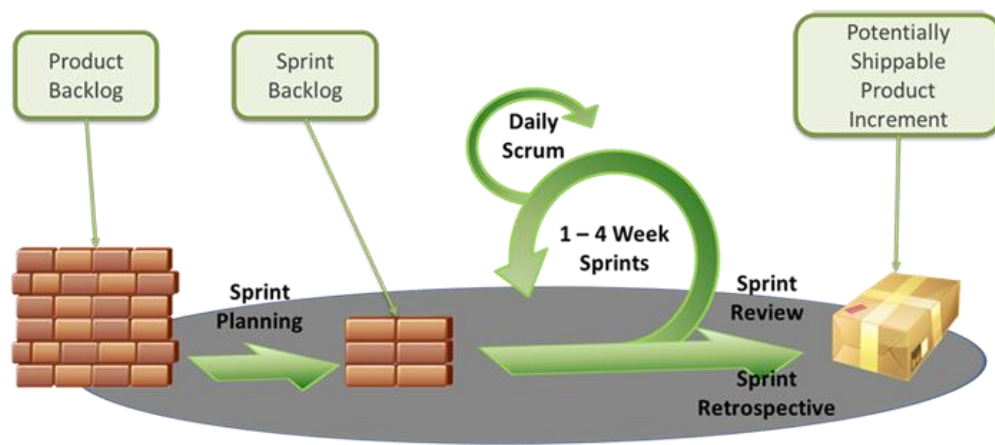


Figura 17-1. Sprint Retrospective

Fuente: (toolshero.nl, <http://www.toolshero.nl/project-management/scrum-agile/>)

1.9 Terminología Médica

1.9.1 Enfermedad

Según la OMS define enfermedad como “Alteración o desviación del estado fisiológico en una o varias partes del cuerpo, por causas en general conocidas, manifestada por síntomas y signos característicos, y cuya evolución es más o menos previsible”. (Universidad De Córdoba, 2015)

1.9.2 Codificación de enfermedades

Una clasificación de enfermedades puede definirse como “Sistema de categorías a las cuales se les asignan entidades morbosas de acuerdo con criterios establecidos” (Ministerio De Sanidad, Servicios Sociales E Igualdad)

Tabla 1-1: Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud.

Cap.	Códigos	Título
I	A00-B99	Ciertas enfermedades infecciosas y parasitarias
II	C00-D48	Neoplasias
III	D50-D89	Enfermedades de la sangre y de los órganos hematopoyéticos y otros trastornos que afectan el mecanismo de la inmunidad
IV	E00-E90	Enfermedades endocrinas, nutricionales y metabólicas

V	F00-F99	Trastornos mentales y del comportamiento
VI	G00-G99	Enfermedades del sistema nervioso
VII	H00-H59	Enfermedades del ojo y sus anexos
VIII	H60-H95	Enfermedades del oído y de la apófisis mastoides
IX	I00-I99	Enfermedades del sistema circulatorio
X	J00-J99	Enfermedades del sistema respiratorio
XI	K00-K93	Enfermedades del aparato digestivo
XII	L00-L99	Enfermedades de la piel y el tejido subcutáneo
XIII	M00-M99	Enfermedades del sistema osteomuscular y del tejido conectivo
XIV	N00-N99	Enfermedades del aparato genitourinario
XV	O00-O99	Embarazo, parto y puerperio
XVI	P00-P96	Ciertas afecciones originadas en el periodo perinatal
XVII	Q00-Q99	Malformaciones congénitas, deformidades y anomalías cromosómicas
XVIII	R00-R99	Síntomas, signos y hallazgos anormales clínicos y de laboratorio, no clasificados en otra parte
XIX	S00-T98	Traumatismos, envenenamientos y algunas otras consecuencias de causa externa
XX	V01-Y98	Causas externas de morbilidad y de mortalidad
XXI	Z00-Z99	Factores que influyen en el estado de salud y contacto con los servicios de salud
XXII	U00-U99	Códigos para situaciones especiales

Fuente: (eciemaps.mspsi, http://eciemaps.mspsi.es/ecieMaps/browser/index_10_2008.html)

1.9.3 Diagnóstico

Procedimiento por el cual se identifica una enfermedad. En términos de la práctica médica, el diagnóstico es un juicio clínico sobre el estado psicofísico de una persona; representa una manifestación en respuesta a una demanda para determinar tal estado.

Proceso diagnóstico

El diagnóstico médico establece a partir de síntomas, signos y los hallazgos de exploraciones complementarias, qué enfermedad padece una persona. Generalmente una enfermedad no está relacionada de una forma exclusiva con un síntoma, es decir, un síntoma no es exclusivo de una enfermedad. Cada síntoma o hallazgo en una exploración presenta una probabilidad de aparición en cada enfermedad.

Herramientas diagnósticas

Síntomas

Son las experiencias subjetivas negativas físicas que refiere el paciente, recogidas por el médico en la anamnesis durante la entrevista clínica, con un lenguaje médico, es decir comprensible para todos los médicos. Por ejemplo, los pacientes a la sensación de falta de aire o percepción incómoda y desagradable en la respiración (disnea), lo llaman ahogo, angustia, fatiga, cansancio fácil.

Signos

Son los hallazgos objetivos que detecta el médico observando al paciente, por ejemplo, la taquipnea a más de 30 respiraciones por minuto. La semiótica médica o semiología clínica es la parte de la medicina que trata de los signos de las enfermedades desde el punto de vista del diagnóstico y del pronóstico.

Exploración física o Semiotécnica

Consiste en diversas maniobras que realiza el médico sobre el paciente, siendo las principales la inspección, palpación, percusión, oler y auscultación, con las que se obtienen signos clínicos más específicos.

Todos los síntomas referidos en la anamnesis y los signos objetivados en la exploración física son anotados en la historia clínica del paciente.

Premisas y pasos necesarios en el diagnóstico

Relación médico-paciente

La relación médico-paciente es vital para obtener la información necesaria que se necesita en el proceso del diagnóstico, además de ser la piedra angular sobre la que descansan las potencialidades terapéuticas del médico. Garantizada esta sinergia, se consigue dar tranquilidad, seguridad y obtener una mejor exposición de los síntomas por parte del enfermo con una mayor cooperación en el examen físico. Hay un principio fundamental en la relación entre el galeno y el doliente, y es hacerle saber a este último que se comprende, y que el médico es capaz de ponerse en su lugar, de ser solidarios con él, y asegurarle que nunca será abandonado.

Interrogatorio

El interrogatorio es la herramienta fundamental para determinar los problemas de salud de los pacientes. Del 50 al 75 por ciento de los diagnósticos se hacen por el interrogatorio. Resulta

esencial identificar la queja o motivo principal por el que el paciente solicita atención médica y describir sus síntomas más importantes. Con solo esta descripción puede bastar para realizar un diagnóstico. No es lo mismo decir que una persona tiene un dolor, que describir también su localización e irradiación, la intensidad, modo de comienzo y de alivio, síntomas acompañantes, etc. Se ha dicho, con toda razón, que un buen interrogatorio “exige la estrategia de un diplomático y el tacto de un confesor”.

Examen físico

El examen físico complementa al interrogatorio. Los signos físicos son las “marcas” objetivas y verificables de la enfermedad y representan hechos sólidos e indiscutibles. Su significado es mayor cuando confirman un cambio funcional o estructural ya sugerido por el interrogatorio.

Es capital la descripción minuciosa de cada signo encontrado. No es lo mismo decir que el paciente tiene una “tumoración”, que describir su ubicación, forma, tamaño, movilidad y sensibilidad. Con esa descripción se afirma si es algo importante o banal, maligno o benigno.

La habilidad en el examen físico se adquiere con la experiencia, pero no solamente es la técnica la que determina el éxito en detectar signos, ni basta con ojos y oídos agudos y entrenados, o dedos especialmente sensibles; se necesita una mente preparada para percatarse de ellos. La pericia en el diagnóstico físico no solo refleja una manera de hacer sino fundamentalmente una manera de pensar.

1.9.4 Tratamiento

Un tratamiento es un conjunto de medios que se utilizan para aliviar o curar una enfermedad, llegar a la esencia de aquello que se desconoce o transformar algo. El concepto es habitual en el ámbito de la medicina.

En función de la patología que sufra un paciente, de la gravedad de la misma y del estado de avance en el que se encuentre el médico pertinente le establecerá un tratamiento u otro. No obstante, entre los más conocidos se encuentran la cirugía, la acupuntura, la ortopedia, el reposo o la hidroterapia.

En el caso del cáncer, existen dos tratamientos fundamentales: la quimioterapia y la radioterapia. Ambas formas suelen ser efectivas en la mayoría de los casos pues son bastante “agresivas”. Esto lo que da lugar es a que además el enfermo que se someta a cualquiera de las dos tenga que hacer frente a duros efectos como pueden ser la pérdida del cabello, náuseas, cansancio o incluso inflamación. (Merino,et.al, 2013)

1.9.5 Historia Clínica

“Es un documento médico legal que consigna la exposición detallada y ordenada de todos los datos relativos a un paciente o usuario, incluye la información del individuo y sus familiares, de los antecedentes, estado actual y evolución, además de los procedimientos y de los tratamientos recibidos”. (Ministerio de Salud Pública, 2007)

1.9.6 Nota de evolución o motivo de consulta

En el motivo de consulta se da un periodo de comunicación directa y franca, en el que el paciente transmite su situación o dudas, donde da sus explicaciones y se relaja, y el especialista realiza una escucha activa, creando un clima de serenidad y seguridad, para captar sus necesidades. (Jennifer)

No se debe confundir motivo de consulta con problema de salud. El primero es la razón de atención sanitaria solicitada por parte del paciente y registrado con sus palabras. Mientras que el problema de salud y el diagnóstico médico son el producto del método clínico que aplica el médico para asignarle un nombre (generalmente normalizado en alguna clasificación profesional) al padecimiento que sufre el enfermo.

CAPÍTULO II

2. MARCO METODOLÓGICO

2.1 Introducción

Este capítulo describe los pasos que se ejecutaron para la realización del proyecto de investigación como tal. Se estudió el motor de base de datos MariaDB, el mismo que al ser un remplazo de MySQL posee más funcionalidades y mejor rendimiento. El lenguaje de programación empleado es Java, para la aplicación web se utiliza el JDK de Oracle, así como para la aplicación móvil el SDK de Android. Además, se muestra el proceso de la metodología SCRUM aplicado en el desarrollo del sistema para la gestión de fichas médicas del Hospital Básico” San Marcos”.

El sistema está formado por cuatro sprints o iteraciones, veinticuatro historias de usuario y veinticuatro pruebas de aceptación, además se consideró necesario para el desarrollo del sistema realizar los veinticuatro casos de uso con sus respectivos diagramas.

2.2 Planificación

Este capítulo explica el desarrollo del proyecto como tal, aquí se describen fase por fase la metodología de desarrollo de Software Scrum, que es la que se empleó en este trabajo.

2.2.1 Usuarios del Sistema

Los actores o usuarios del sistema son: Administrador, doctor, enfermera y paciente y sus permisos se definen en el Tabla 1-2.

Tabla 1-2: Usuarios del Sistema Médico

Usuario	Permisos
Administrador	Es la persona que puede ingresar, actualizar o eliminar los datos de los usuarios del sistema. También es quién gestiona los menús de la página web.
Doctor	Es la persona que puede ingresar las historias clínicas. Puede registrar el motivo, diagnóstico y tratamiento referente a un paciente.
Enfermera	Es la persona que puede otorgar una cita médica a los pacientes.

	Puede hacer el ingreso de los datos personales de los pacientes en general. Registra la toma de los signos vitales de cada paciente.
Paciente	Es la persona que puede hacer la reserva, modificación y cancelación de una cita médica. Además, puede ingresar sus datos personales. Puede visualizar la información sobre las consultas médicas reservadas.

Realizado por: Fausto Orozco, 2017.

2.2.2 Recopilación de los requerimientos para el desarrollo del Sistema

Los requerimientos del sistema son proporcionados por el Product owner generalmente de forma oral en una entrevista, los mismos que fueron organizados de la siguiente manera:

1. El sistema permitirá la autenticación de usuarios
2. El sistema permitirá ingresar, modificar y eliminar a los usuarios del sistema.
3. El sistema permitirá ingresar, modificar y eliminar a los horarios de atención en el sistema.
4. El sistema permitirá ingresar, modificar y eliminar la asignación de horarios de los médicos en el sistema.
5. El sistema permitirá ingresar y modificar los datos personales de un paciente.
6. El sistema permitirá actualizar la foto personal de un paciente.
7. El sistema permitirá asignar, modificar y cancelar una cita médica para un determinado paciente.
8. El sistema permitirá llenar la historia clínica de un paciente.
9. El sistema permitirá redactar el motivo de consulta, diagnóstico y tratamiento de un paciente.
10. El sistema permitirá obtener el reporte de la historia clínica de un paciente.
11. El sistema permitirá ingresar y modificar los datos de la receta médica del paciente.
12. El sistema permitirá obtener el reporte de la receta médica de un paciente.
13. El sistema permitirá registrar los datos de los signos vitales de un paciente.
14. El sistema permitirá obtener el listado de los doctores que pertenecen al Hospital básico San Marcos.
15. El sistema permitirá ingresar, modificar y eliminar los datos de una clínica en el sistema.
16. El sistema permitirá ingresar, modificar y eliminar los datos de una especialidad en el sistema.
17. El sistema permitirá ingresar, modificar y eliminar los datos de un consultorio en el sistema.
18. El sistema permitirá ingresar, modificar y eliminar los datos de un doctor en el sistema.
19. El sistema permitirá ingresar, modificar y eliminar los datos de una enfermedad en el sistema.

20. El sistema permitirá obtener el total de pacientes atendidos en un determinado día con su respectivo listado
21. El sistema permitirá solicitar, modificar y cancelar una cita médica utilizando un dispositivo móvil.

2.2.3 Historias de usuario

El sistema está formado por veinticuatro historias de usuario de acuerdo a los requerimientos específicos que se recopiló. Las historias de usuario son los requerimientos del sistema organizados de acuerdo a cada usuario del sistema.

HU1. Como Usuario del sistema pretendo autenticarme.

ADMINISTRADOR

HU2. Como Administrador pretendo ingresar, modificar y eliminar a los usuarios del sistema.

HU3. Como Administrador pretendo ingresar, modificar y eliminar a los días de la semana en los que atiende un doctor.

HU4. Como Administrador pretendo ingresar, modificar y eliminar a los horarios de atención en el sistema.

HU5. Como Administrador pretendo ingresar, modificar y eliminar la asignación de horarios de los médicos en el sistema.

DOCTOR

HU6. Como Doctor pretendo llenar la historia clínica de un paciente con la finalidad de conocer sus antecedentes de salud y así dar un diagnóstico y tratamiento adecuado.

HU7. Como Doctor pretendo redactar el motivo de consulta, diagnóstico y tratamiento de un paciente con la finalidad de tener un registro de sus datos, valoraciones e información a lo largo del proceso asistencial.

HU8. Como Doctor pretendo obtener un reporte de la historia clínica de un paciente.

HU9. Como Doctor pretendo prescribir la receta médica de un paciente ingresando los datos de los medicamentos y sus respectivas indicaciones.

HU10. Como Doctor pretendo obtener un reporte de la receta médica de un paciente.

ENFERMERA

HU11. Como Enfermera pretendo ingresar, modificar los datos personales de los pacientes con la finalidad de tener un registro actualizado de los mismos.

HU12. Como Enfermera pretendo actualizar la foto personal de un paciente.

HU13. Como Enfermera necesito asignar, modificar y cancelar una cita médica para un determinado paciente

HU14. Como Enfermera pretendo realizar la toma de los signos vitales de un paciente con la finalidad de ser atendido por el médico.

HU15. Como Enfermera pretendo obtener el listado de los doctores que pertenecen al Hospital básico San Marcos.

HU16. Como Enfermera pretendo ingresar, modificar y eliminar los datos de una clínica en el sistema.

HU17. Como Enfermera pretendo ingresar, modificar y eliminar los datos de una especialidad en el sistema.

HU18. Como Enfermera pretendo ingresar, modificar y eliminar los datos de un consultorio en el sistema.

HU19. Como Enfermera pretendo ingresar, modificar y eliminar los datos de un doctor en el sistema.

HU20. Como Enfermera pretendo ingresar, modificar y eliminar los datos de una enfermedad en el sistema.

HU21. Como Enfermera deseo obtener el total de pacientes atendidos en un determinado día con su respectivo listado.

PACIENTE

HU22. Como paciente pretendo ingresar y modificar mis datos personales

HU23. Como paciente pretendo solicitar, modificar y cancelar una consulta médica a través de la aplicación web

HU24. Como paciente pretendo solicitar, modificar y cancelar una cita médica utilizando un dispositivo móvil.

Product Backlog

En base a las historias de usuarios se realiza el Product Backlog, en donde se ubican las funciones a realizar de acuerdo a la prioridad de cada historia la misma que se encuentra en un rango de 1 a 10, la estimación se la realiza en horas como se muestra en el Tabla 3-2.

Tabla 2-2: Product Backlog

Id	Historia de Usuario	Prioridad	Sprint
HU1	Como Usuario del sistema pretendo autenticarme.	10	1
HU2	Como Administrador pretendo ingresar, modificar y eliminar a los usuarios del sistema.	10	1
HU3	Como Administrador pretendo ingresar, modificar y eliminar a los días de la semana en los que atiende un doctor.	10	1
HU4	Como Administrador pretendo ingresar, modificar y eliminar a los horarios de atención en el sistema.	10	1
HU5	Como Administrador pretendo ingresar, modificar y eliminar la asignación de horarios de los médicos en el sistema.	9	2
HU6	Como Doctor pretendo llenar la historia clínica de un paciente con la finalidad de conocer sus antecedentes de salud y así dar un diagnóstico y tratamiento adecuado.	7	4
HU7	Como Doctor pretendo redactar el motivo de consulta, diagnóstico y tratamiento de un paciente con la finalidad de tener un registro de sus datos, valoraciones e información a lo largo del proceso asistencial.	7	4
HU8	Como Doctor pretendo obtener un reporte de la historia clínica de un paciente.	7	4
HU9	Como Doctor pretendo prescribir la receta médica de un paciente ingresando los datos de los medicamentos y sus respectivas indicaciones.	7	4

HU10	Como Doctor pretendo obtener un reporte de la receta médica de un paciente.	7	4
HU11	Como Enfermera pretendo ingresar y modificar los datos personales de los pacientes y a su vez el sistema le asignará un número de historia clínica al paciente.	9	2
HU12	Como Enfermera pretendo actualizar la foto personal de un paciente.	9	2
HU13	Como Enfermera necesito asignar, modificar y cancelar una cita médica para un determinado paciente	8	3
HU14	Como Enfermera pretendo realizar la toma de los signos vitales de un paciente con la finalidad de ser atendido por el médico.	8	3
HU15	Como Enfermera pretendo obtener el listado de los doctores que pertenecen al Hospital básico San Marcos.	7	4
HU16	Como Enfermera pretendo ingresar, modificar y eliminar los datos de una clínica en el sistema.	10	1
HU17	Como Enfermera pretendo ingresar, modificar y eliminar los datos de una especialidad en el sistema.	10	1
HU18	Como Enfermera pretendo ingresar, modificar y eliminar los datos de un consultorio en el sistema.	9	2
HU19	Como Enfermera pretendo ingresar, modificar y eliminar los datos de un doctor en el sistema.	9	2
HU20	Como Enfermera pretendo ingresar, modificar y eliminar los datos de una enfermedad en el sistema.	9	2
HU21	Como Enfermera deseo obtener el total de pacientes atendidos en un determinado día con	7	4
HU22	Como paciente pretendo ingresar y modificar mis datos personales	8	3
HU23	Como paciente pretendo solicitar, modificar y cancelar una consulta médica a través de la aplicación web	8	3

HU24	Como paciente pretendo solicitar, modificar y cancelar una cita médica utilizando un dispositivo móvil.	8	3
------	---	---	---

Realizado por: Fausto Orozco, 2017

2.2.4 Iteraciones

El sistema está dividido en cuatro sprints o iteraciones, dos sprints de seis historias de usuario, un sprint de cinco historias de usuario y un sprint de siete historias de usuario.

Sprint 1

En este primer sprint se han considerado las historias de usuario más relevantes del sistema

Tabla 3-2: Sprint 1

ID	Tipo de elemento	Actividad	Estimación	Responsable	Estado
HU1	Historia de usuario	Como Usuario del sistema pretendo autenticarme.	20		Completo
	Tarea	Creación de la capa acceso a datos	3	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	5	Fausto Orozco	Completo
	Tarea	Pruebas	7	Fausto Orozco	Completo
HU2	Historia de usuario	Como Administrador pretendo ingresar, modificar y eliminar a los usuarios del sistema.	30		Completo
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	7	Fausto Orozco	Completo
	Tarea	Creación de interfaz	8	Fausto Orozco	Completo
	Tarea	Pruebas	10	Fausto Orozco	Completo

HU3	Historia de usuario	Como Administrador pretendo ingresar, modificar y eliminar a los días de la semana en los que atiende un doctor	20		Completo
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	5	Fausto Orozco	Completo
	Tarea	Pruebas	5	Fausto Orozco	Completo
HU4	Historia de usuario	Como Administrador pretendo ingresar, modificar y eliminar a los horarios de atención en el sistema.	30	Fausto Orozco	Completo
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	15	Fausto Orozco	Completo
	Tarea	Pruebas	5	Fausto Orozco	Completo
HU16	Historia de usuario	Como Enfermera pretendo ingresar, modificar y eliminar los datos de una clínica en el sistema.	20	Fausto Orozco	Completo
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo

	Tarea	Creación de interfaz	5	Fausto Orozco	Completo
	Tarea	Pruebas	5	Fausto Orozco	Completo
HU17	Historia de usuario	Como Enfermera pretendo ingresar, modificar y eliminar los datos de una especialidad en el sistema.	20	Fausto Orozco	Completo
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	5	Fausto Orozco	Completo
	Tarea	Pruebas	5	Fausto Orozco	Completo
Total Horas			140		

Realizado por: Fausto Orozco, 2017

Sprint 2

Tabla 4-2: Sprint 2

ID	Tipo de elemento	Actividad	Estimación	Responsable	Estado
HU20	Historia de usuario	Como Enfermera pretendo ingresar, modificar y eliminar los datos de una enfermedad en el sistema.	40	Fausto Orozco	Completo
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	10	Fausto Orozco	Completo
	Tarea	Creación de interfaz	15	Fausto Orozco	Completo

	Tarea	Pruebas	10	Fausto Orozco	Completo
HU18	Historia de Usuario	Como Enfermera pretendo ingresar, modificar y eliminar los datos de un consultorio en el sistema.	30		
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	7	Fausto Orozco	Completo
	Tarea	Pruebas	13	Fausto Orozco	Completo
HU19	Historia de Usuario	Como Enfermera pretendo ingresar, modificar y eliminar los datos de un doctor en el sistema.	20		
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	5	Fausto Orozco	Completo
	Tarea	Pruebas	5	Fausto Orozco	Completo
HU5	Historia de Usuario	Como Administrador pretendo ingresar, modificar y eliminar la asignación de horarios de los médicos en el sistema.	30		
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	10	Fausto Orozco	Completo

	Tarea	Pruebas	10	Fausto Orozco	Completo
HU11	Historia de Usuario	Como Enfermera pretendo ingresar y modificar los datos personales de los pacientes y a su vez el sistema le asignará un número de historia clínica al paciente.	45		
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	10	Fausto Orozco	Completo
	Tarea	Creación de interfaz	15	Fausto Orozco	Completo
	Tarea	Pruebas	15	Fausto Orozco	Completo
HU12	Historia de Usuario	Como Enfermera pretendo actualizar la foto personal de un paciente.	20		
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	5	Fausto Orozco	Completo
	Tarea	Pruebas	5	Fausto Orozco	Completo
Total Horas			185		

Realizado por: Fausto Orozco, 2017

Sprint 3

Tabla 5-2: Sprint 3

ID	Tipo de elemento	Actividad	Estimación	Responsable	Estado
----	------------------	-----------	------------	-------------	--------

HU13	Historia de usuario	Como Enfermera necesito asignar, modificar y cancelar una cita médica para un determinado paciente	45	Fausto Orozco	Completo
	Tarea	Creación de la capa acceso a datos	10	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	15	Fausto Orozco	Completo
	Tarea	Pruebas	15	Fausto Orozco	Completo
HU22	Historia de usuario	Como paciente pretendo ingresar y modificar mis datos personales.	20	Fausto Orozco	Completo
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	5	Fausto Orozco	Completo
	Tarea	Pruebas	5	Fausto Orozco	Completo
HU23	Historia de Usuario	Como paciente pretendo solicitar, modificar y cancelar una consulta médica a través de la aplicación web	45		
	Tarea	Creación de la capa acceso a datos	10	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	15	Fausto Orozco	Completo

	Tarea	Pruebas	15	Fausto Orozco	Completo
HU24	Historia de Usuario	Como paciente pretendo solicitar, modificar y cancelar una cita médica utilizando un dispositivo móvil.	60		
	Tarea	Creación de la capa acceso a datos	15	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	15	Fausto Orozco	Completo
	Tarea	Creación de interfaz	15	Fausto Orozco	Completo
	Tarea	Pruebas	15	Fausto Orozco	Completo
HU14	Historia de Usuario	Como Enfermera pretendo realizar la toma de los signos vitales de un paciente con la finalidad de ser atendido por el médico.	20		
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	5	Fausto Orozco	Completo
	Tarea	Pruebas	5	Fausto Orozco	Completo
Total Horas			190		

Realizado por: Fausto Orozco, 2017

Sprint 4

Tabla 6-2: Sprint 4

ID	Tipo de elemento	Actividad	Estimación	Responsable	Estado
----	------------------	-----------	------------	-------------	--------

HU6	Historia de usuario	Como Doctor pretendo llenar la historia clínica de un paciente con la finalidad de conocer sus antecedentes de salud y así dar un diagnóstico y tratamiento adecuado.	35	Fausto Orozco	Completo
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	10	Fausto Orozco	Completo
	Tarea	Pruebas	15	Fausto Orozco	Completo
HU7	Historia de usuario	Como Doctor pretendo redactar el motivo de consulta, diagnóstico y tratamiento de un paciente con la finalidad de tener un registro de sus datos, valoraciones e información a lo largo del proceso asistencial.	30	Fausto Orozco	Completo
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	5	Fausto Orozco	Completo
	Tarea	Pruebas	15	Fausto Orozco	Completo
HU8	Historia de Usuario	Como Doctor pretendo obtener un reporte de la	50		

		historia clínica de un paciente.			
	Tarea	Creación de la capa acceso a datos	10	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	15	Fausto Orozco	Completo
	Tarea	Creación de interfaz	10	Fausto Orozco	Completo
	Tarea	Pruebas	15	Fausto Orozco	Completo
HU15	Historia de Usuario	Como Enfermera pretendo obtener el listado de los doctores que pertenecen al Hospital Básico San Marcos.	25		
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	5	Fausto Orozco	Completo
	Tarea	Pruebas	10	Fausto Orozco	Completo
HU21	Historia de Usuario	Como Enfermera deseo obtener el total de pacientes atendidos en un determinado día con su respectivo listado.	25		
	Tarea	Creación de la capa acceso a datos	5	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	5	Fausto Orozco	Completo
	Tarea	Creación de interfaz	5	Fausto Orozco	Completo
	Tarea	Pruebas	10	Fausto Orozco	Completo

HU9	Historia de usuario	Como Doctor pretendo prescribir la receta médica de un paciente ingresando los datos de los medicamentos y sus respectivas indicaciones.	60	Fausto Orozco	Completo
	Tarea	Creación de la capa acceso a datos	10	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	10	Fausto Orozco	Completo
	Tarea	Creación de interfaz	25	Fausto Orozco	Completo
	Tarea	Pruebas	15	Fausto Orozco	Completo
HU10	Historia de Usuario	Como Doctor pretendo obtener un reporte de la receta médica de un paciente.	50		
	Tarea	Creación de la capa acceso a datos	10	Fausto Orozco	Completo
	Tarea	Creación de la capa lógica de negocios	15	Fausto Orozco	Completo
	Tarea	Creación de interfaz	10	Fausto Orozco	Completo
	Tarea	Pruebas	15	Fausto Orozco	Completo
Total Horas			275		

Realizado por: Fausto Orozco, 2017

2.3 Diseño

2.3.1 Arquitectura del sistema

La arquitectura del sistema para la Gestión de Fichas Médicas del Hospital Básico “San Marcos” está formada por dos escenarios, en el primer caso la solución es una aplicación web JSP y para el segundo caso la solución es una aplicación móvil Android, utilizando para cada uno de estos escenarios una arquitectura cliente-servidor en tres capas, estas son la capa de presentación, la

capa de negocios y la capa de persistencia (acceso a la base de datos), porque se requiere que la información de la entidad de salud se encuentre almacenada en un servidor de base de datos MariaDB externo. Para una mejor explicación de la arquitectura, a continuación, se muestra a través de los diagramas de componentes y despliegue.

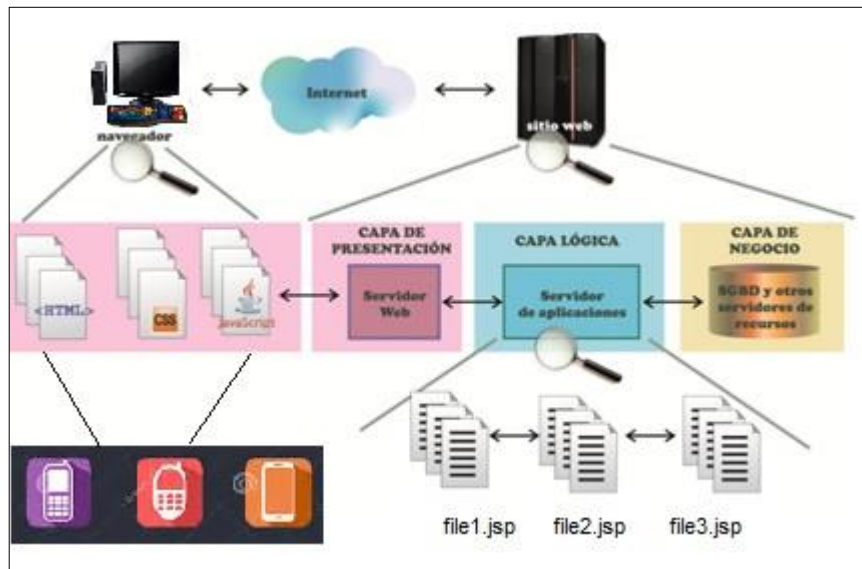


Figura 1-2. Capas del Sistema Gestión de Fichas Médicas

Fuente: Fausto Orozco, 2017

Capa de presentación: denominada capa de usuario se puede visualizar de forma real al sistema, comunica y captura información en procesos mínimos. También es conocida como la interfaz de usuario la cual debe ser amigable y de fácil uso, esta capa tiene una comunicación única directa con la capa de negocio. (Ruiz, 2013)

Capa de negocio: se encuentran los programas que se ejecutan, se envían peticiones y se reciben respuestas tras un debido proceso, en esta capa se establecen las reglas que se deben cumplir. La relación con la capa de presentación permite recibir las solicitudes y mostrar los resultados, mientras con la capa de persistencia se logra solicitar al gestor de base de datos guardar u obtener datos. (Ruiz, 2013)

Capa de persistencia: se encarga de proporcionar una biblioteca de funcionalidades para acceder a la base de datos, recibe solicitudes de almacenamiento o recuperación desde la capa de negocio. (Ruiz, 2013)

2.3.2 Diagrama de componentes

Para mostrar cada uno de los componentes que integran el sistema se lo realiza a través del diagrama de componentes que se lo puede observar en la Figura 3-2.

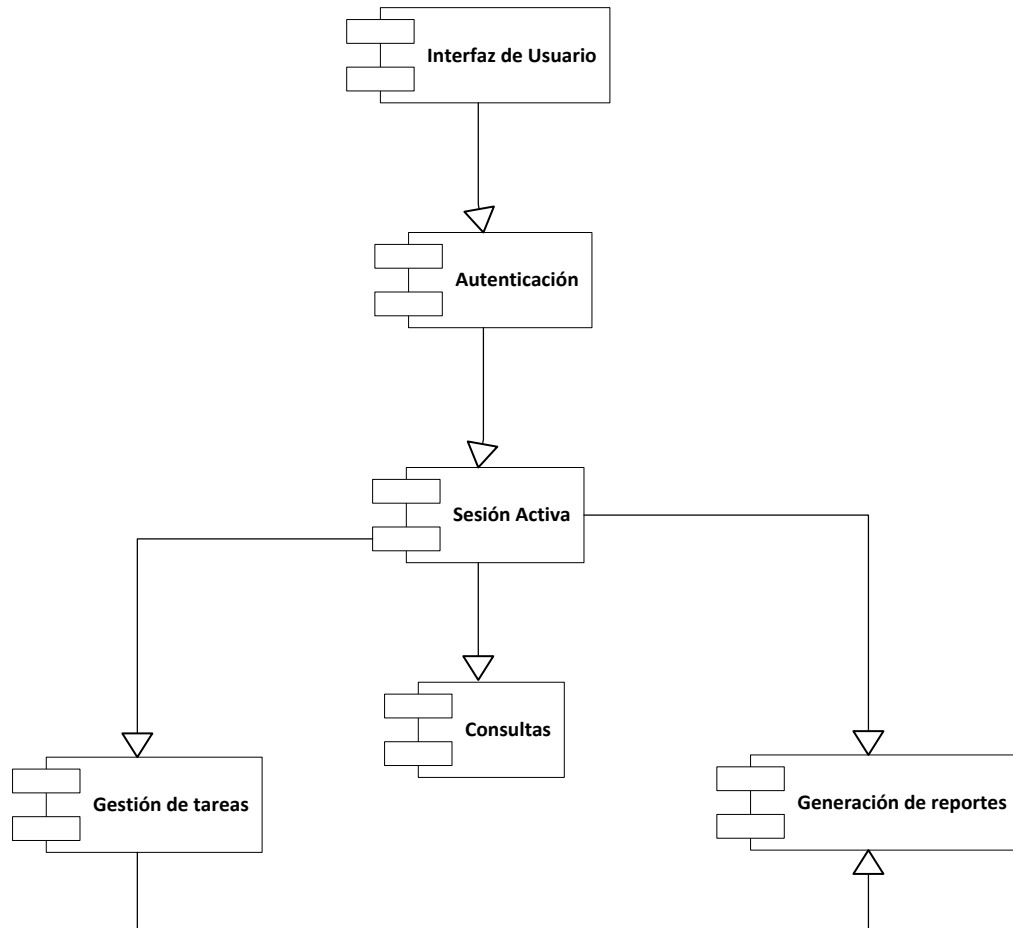


Figura 2-2. Diagrama de Componentes

Fuente: Fausto Orozco, 2017

2.3.3 Diagrama de despliegue

En el desarrollo e implementación del sistema, integrando a JSP y MariaDB como DBMS para la gestión de fichas médicas del Hospital Básico “San Marcos” utilizando tecnología móvil, como se puede observar en la Figura 4-2, se tiene una representación de los elementos básicos que forman parte de la arquitectura del mismo.

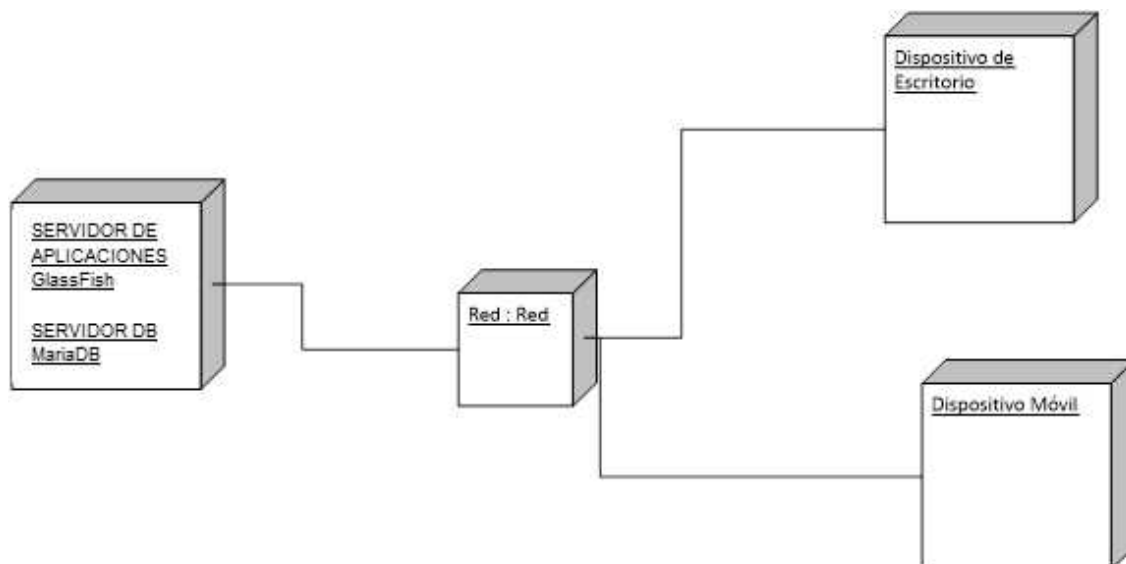


Figura 3-2. Diagrama de Despliegue

Fuente: Fausto Orozco, 2017

El sistema está compuesto por un servidor de aplicaciones web Glassfish para la aplicación JSP, además de un servidor de base de datos MariaDB donde se almacenará los datos y la información de los pacientes del Hospital Básico San Marcos, además se podrá acceder desde cualquier dispositivo de escritorio al sitio web JSP y finalmente el acceso desde cualquier dispositivo móvil a la aplicación Android.

2.3.4 Esquema de la base de datos

Como parte fundamental para el desarrollo de la aplicación es necesario contar con el diseño de la base de datos el cual se lo obtiene mediante las historias de usuario, mediante este diseño se puede visualizar la estructura de la base de datos a través del modelo relacional.

Diagrama Modelo Relacional

Este Diagrama se encuentra representado por entidades, atributos y sus respectivas relaciones que se utilizarán para la gestión de la información del Hospital Básico San Marcos para una mejor perspectiva y análisis se puede observar en el Anexo C.

Diccionario de datos

El diccionario de datos es un conjunto de metadatos que contiene las características lógicas y puntuales de los datos que se van a utilizar en el desarrollo del sistema de gestión de fichas médicas del Hospital Básico San Marcos, el mismo que se puede observar en el Anexo D.

2.3.5 Casos de Uso

Los casos de uso manifiestan el ambiente en que la aplicación interactúa con el usuario u otra aplicación y se los realiza en base a cada historia de usuario.

El sistema está formado por veinticuatro casos de uso de acuerdo al número de historias de usuario que se elaboró. A continuación, se detallan los tres primeros casos de uso del sistema, para un detalle de los restante casos de uso ver el Anexo E.

Tabla 7-2: HU-01: Como Usuario del sistema pretendo autenticarme.

Identificador caso de uso	R-AUTENTICA_USUARIO
Nombre del caso de uso	Autenticar usuario.
Actores	Usuario
Propósito	Ingresar al sistema.
Visión general	El usuario debe autenticarse para ingresar al sistema.
Tipo	Primario y esencial
Referencias	HU-01
Curso Típico de Eventos	
Usuario	Aplicación
1.- Presionar el botón Sitio Usuarios (Doctores, Enfermeras, Pacientes o Administrador).	2.- Muestra en la pantalla el formulario para ingresar los datos de sesión.
3.-Ingresar los datos de: 1. Usuario 2. Contraseña	4.- Valida los datos ingresados
5.- Presionar el botón aceptar.	6.-Muestra en pantalla la página del menú Sitio Usuarios (Doctores, Enfermeras, Pacientes o Administrador).
Cursos Alternativos	
Línea 3. Si el usuario o contraseña están mal ingresados o son erróneos se visualizará un mensaje que indica que hubo un ingreso fallido.	

Realizado por: (Fausto Orozco, 2017)

Tabla 8-2: HU-02: Como Administrador pretendo ingresar, modificar y eliminar a los usuarios del sistema.

Identificador caso de uso	R-REGISTRA_USUARIO
Nombre del caso de uso	Registrar usuario.
Actores	Administrador
Propósito	Ingresar los datos de los usuarios al sistema.
Visión general	El administrador es la persona encargada de gestionar los datos de los usuarios del sistema.
Tipo	Primario y esencial
Referencias	HU-02
Curso Típico de Eventos	
Administrador	Aplicación
1.- Seleccionar del menú Usuario, la opción “Listado Usuarios”.	2.- Muestra un formulario con el listado de usuarios registrados en el sistema.
3.- Presionar el botón Nuevo Usuario.	4.- Muestra en el centro de la pantalla un campo para ingresar el dato cédula del usuario.
5.-Ingresar el dato de: 1. Cédula	6.- Valida el dato ingresado
7.- Presionar el botón buscar.	8.- Muestra el formulario para ingresar los datos personales del usuario.
9.- Ingresar datos del usuario	10.- Valida los datos ingresados
11.- Presionar el botón insertar.	12.-Presentar mensaje de datos insertados correctamente.
Cursos Alternativos	
Línea 5. Si ha ingresado el dato erróneamente se visualizará un mensaje dato incorrecto, por lo tanto, tiene que volver a ingresar un dato correcto.	
Línea 7. Si no ha ingresado todos los datos solicitados se visualizará un mensaje que indica que todos los campos son obligatorios.	

Realizado por: (Fausto Orozco, 2017)

Tabla 9-2: HU-03: Como Administrador pretendo ingresar, modificar y eliminar a los días de la semana en los que atiende un doctor.

Identificador caso de uso	R-REGISTRA_DIA_SEMANA
Nombre del caso de uso	Registrar día de la semana.
Actores	Administrador
Propósito	Ingresar los datos de los días de la semana.
Visión general	El administrador es la persona encargada de gestionar los datos de los días de la semana en los que atiende un doctor.
Tipo	Primario y esencial
Referencias	HU-03
Curso Típico de Eventos	
Administrador	Aplicación
1.- Seleccionar del menú Ingresos, la opción “Lista días”	2.- Muestra en pantalla una tabla con los días de la semana.
3.- Presionar el botón Nuevo día	4.- Muestra el formulario para ingresar los datos de un día.
5.-Ingresar los datos de: 1. Nombre	6.- Valida los datos ingresados
7.- Presionar el botón insertar.	8.-Presentar mensaje de datos insertados correctamente.
Cursos Alternativos	
Línea 5. Si no ha ingresado todos los datos solicitados se visualizará un mensaje que indica que todos los campos son obligatorios.	

Realizado por: (Fausto Orozco, 2017)

2.3.6 Diagramas de Casos de Usos

En los diagramas de casos de uso se definen los actores que interactúan con el sistema, al igual que sus relaciones e iteraciones. Esto permitirá tener una mejor visión de lo que se realiza en el sistema y como se lo realiza.

El sistema está formado por veinticuatro diagramas de casos de uso de acuerdo al número de historias de usuario que se obtuvo. A continuación, se detallan los tres primeros diagramas de casos de uso del sistema, para un detalle de los restante casos de uso ver el Anexo F.

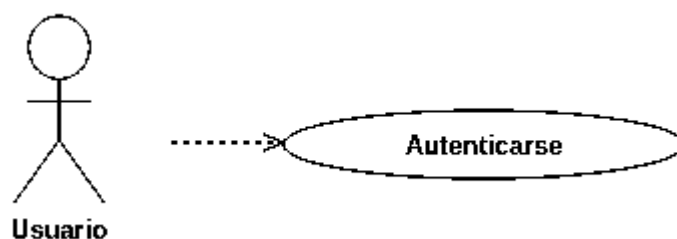


Figura 4-2. Caso de Uso HU01.

Realizado por: (Fausto Orozco, 2017)

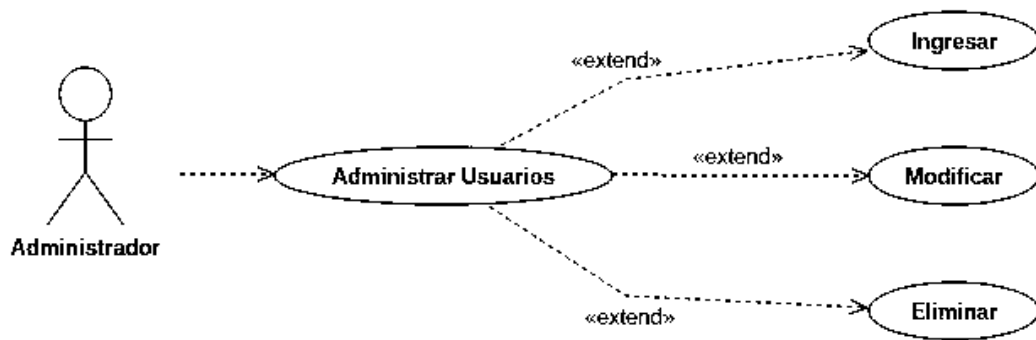


Figura 5-2. Caso de Uso HU02.

Realizado por: (Fausto Orozco, 2017)

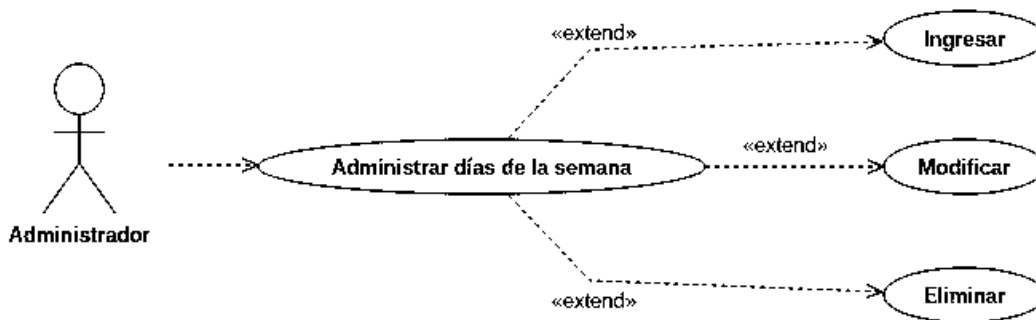


Figura 6-2. Caso de Uso HU03.

Realizado por: (Fausto Orozco, 2017)

2.4 Codificación o Desarrollo

2.4.1 Conexión de MariaDB con Android

En el desarrollo de la aplicación Android, enfatizamos el cómo lograr conectarnos a la base de datos MariaDB, la funcionalidad de la aplicación mobile es la de permitir hacer la reserva de citas médicas con el médico de preferencia del usuario, y para alcanzar este objetivo se necesita de varias de las operaciones como son: consultar datos del paciente o registrar datos del paciente, consultar datos de las clínicas, especialidades, consultorios y doctores, consultar las reservas de citas médicas de un determinado paciente y otros.

Para conectarnos desde una aplicación mobile hacia una base de datos el SDK de Android, nos proporciona una clase especial llamada AsyncTask, la misma que permite realizar este procedimiento de manera asíncrona.

A continuación, mostramos el código fuente que permite consultar los datos de un paciente dado su número de cédula.

```
public class MostrarDatosPacienteAsync extends AsyncTask<String, Float, Object[]> {

    @Override

    protected Object[] doInBackground(String... strings) {

        Object[] arrayObjects = new Object[2];

        // ##### OBTENER LOS DATOS DEL PACIENTE #####

        Paciente ubPaciente = null;

        HistoriaClinica ubHistoriaClinica = null;

        try {

            System.out.println("CEDULA DEL PACIENTE: "+strings[0]);

            ubPaciente =Paciente.sp_obtener_paciente_dado_cedula(strings[0]);

            Integer id_paciente =ubPaciente.getId_paciente();

            ubHistoriaClinica =HistoriaClinica.ObtenerHCdadoPaciente(id_paciente);

        } catch (Exception ex) {

            System.out.println("Error - Obtener Datos Paciente: " + ex.getMessage());

        }

        arrayObjects[0] = ubPaciente;

        arrayObjects[1] = ubHistoriaClinica;

        return arrayObjects;

    }

    @Override

    protected void onPostExecute(Object[] obj) {

        Paciente ubPaciente = (Paciente)obj[0];

        HistoriaClinica ubHistoriaClinica = (HistoriaClinica)obj[1];

        lblValorNombres.setText(ubPaciente.getNombrespac());

        lblValorApellidos.setText(ubPaciente.getApellidospac());

        lblValorCedula.setText(ubPaciente.getCipaciente());

        lblValorNumHistoria.setText(Integer.toString(ubHistoriaClinica.getNumhistoria()));

        lblValorFechaNacimiento.setText(ubPaciente.getFechanac());

    }

}
```

```

        lblValorEmail.setText(ubPaciente.getEmail());
    }
}

```

Como se puede ver en este trozo de código java, así es como se logra obtener los datos del paciente dado su cédula. A continuación, se describe los pasos a seguir:

- Primeramente, se crea la clase *MostrarDatosPacienteAsync* la misma que se hereda de nuestra clase llamada *AsyncTask*.
- *AsyncTask* tiene definido algunos métodos previamente, y en este caso se tendrá que sobrescribir varias de las funciones para que nos permita hacer una conexión asíncrona hacia la base de datos.
- Se pasa un parámetro a la clase *MostrarDatosPacienteAsync*, específicamente al método *doInBackground*, este parámetro es un array de *String* llamado *strings*.
- Ahora procedemos a obtener los datos del paciente pasando como parámetro el valor de *strings[0]* al método *sp_obtener_paciente_dado_cedula* de la clase *Paciente*.
- Después obtenemos los datos de la clase *HistoriaClinica*, ahora pasando como parámetro al método *ObtenerHCdadoPaciente* el valor del *IdPaciente*, es decir el *Id* del paciente.
- El método *doInBackground* retorna los dos objetos *ubPaciente* y *ubHistoriaClinica*, que previamente fueron asignados a *arrayObjects[0]* y *arrayObjects[1]* respectivamente, de esta manera se retorna el array *arrayObjects*, el mismo que contiene los dos objetos.
- Por último, en el método *onPostExecute* se recibe el array *arrayObjects* retornado por el método *doInBackground*, y es aquí donde se asigna los datos del paciente y de historia clínica a sus respectivos elementos *TextView*, para que puedan ser visualizados por el usuario.

2.4.2 Conexión de MariaDB con JSP

En el desarrollo de la aplicación web, muestro como se puede conectar a la base de datos MariaDB, una de las funcionalidades de la aplicación es la de permitir hacer la reserva de citas médicas con el médico de preferencia del usuario, y para alcanzar este objetivo se necesita de varias de las operaciones como son: consultar datos del paciente o registrar datos del paciente, consultar datos de las clínicas, especialidades, consultorios y doctores, consultar las reservas de citas médicas de un determinado paciente y otros.

Para conectarnos desde una aplicación web hacia una base de datos se utiliza el JDBC de Oracle, ya que este provee de clases con sus respectivos métodos para lograr con éxito abrir una conexión hacia MariaDB.

A continuación, mostramos el código fuente que permite consultar los datos de un paciente dado su número de cédula.

```
String cedula = request.getParameter("txtCedula");
ubPaciente = Paciente.sp_obtener_paciente_dado_cedula(cedula);
Integer id_paciente = ubPaciente.getId_paciente();
ubHistoriaClinica = HistoriaClinica.ObtenerHCdadoPaciente(id_paciente);

out.println(ubPaciente.getNombrespac());
out.println(ubPaciente.getApellidospac());
out.println(ubPaciente.getDireccion());
out.println(ubPaciente.getFechanac());
out.println(ubPaciente.getEmail());
```

- El primer paso es solicitar al usuario que ingrese el número de cédula, y con este procedemos a obtener los datos del paciente pasando como parámetro el valor de cédula al método *sp_obtener_paciente_dado_cedula* de la clase Paciente.
- Después obtenemos los datos de la clase HistoriaClinica, ahora pasando como parámetro al método *ObtenerHCdadoPaciente* el valor del IdPaciente, es decir el Id del paciente.
- Por último, se obtiene los datos del paciente y de historia clínica y estos pueden ser visualizados por el usuario utilizando el método *out.println*.

2.5 Pruebas

2.5.1 Pruebas de Aceptación

Las pruebas de aceptación, también conocida como prueba de aceptación del usuario, es un tipo de ensayos se realiza con el fin de verificar si el producto ha sido desarrollado de acuerdo con las normas y criterios establecidos y cumple con todos los requisitos especificados por el Product owner. Este tipo de pruebas se lleva a cabo generalmente por un usuario/cliente.

El sistema está formado por veinticuatro pruebas de aceptación de acuerdo al número de historias de usuario que se obtuvo. A continuación, se detallan las tres primeras historias de usuario del sistema, para un detalle de las restante historias de usuario ver el Anexo G.

HU1. Como Usuario del sistema pretendo autenticarme.

Descripción: Para acceder al sistema cada uno de los usuarios tales como: Doctor, enfermera y paciente deben autenticarse, los cuales visualizarán su respectivo menú a utilizar.

Condiciones de ejecución: El Doctor, enfermera y paciente ingresará su usuario y contraseña para autenticarse y acceder al sistema.

Parámetros de entrada

- Usuario y Contraseña.

Resultado esperado: El usuario accede al sistema.

Evaluación de la prueba: Prueba satisfactoria.

HU2. Como Administrador pretendo ingresar, modificar y eliminar a los usuarios del sistema.

Descripción: Quién podrá ingresar, modificar y eliminar los datos de los usuarios del sistema será el Administrador.

Condiciones de ejecución: Dependiendo de las funciones que le corresponden a cada usuario, el administrador debe asignar los respectivos permisos de acuerdo a los roles de los usuarios.

Parámetros de entrada

- El administrador ingresará los datos de los usuarios, el sistema los validará.
- El administrador modificará los datos de los usuarios, el sistema los validará.
- El administrador eliminará los datos de los usuarios, se confirmará la eliminación.

Resultado esperado: Se almacena la información en la base de datos, en los casos de ingreso y modificación. De aceptarse la eliminación, los datos serán eliminados de la base.

Evaluación de la prueba: Prueba satisfactoria.

HU3. Como Administrador pretendo ingresar, modificar y eliminar a los días de la semana en los que atiende un doctor.

Descripción: El Administrador podrá ingresar, modificar y eliminar los días de la semana en los que atiende un doctor de acuerdo con sus horarios de trabajo.

Condiciones de ejecución: De acuerdo a los horarios establecidos para los doctores el administrador podrá ingresar, modificar y eliminar los días de atención de un médico a los pacientes.

Parámetros de entrada

- El administrador ingresará los días de atención de un doctor, el sistema valida la información.
- El administrador modificará los días de atención de un doctor, el sistema valida la información.
- El administrador eliminará los días de atención de un doctor, se confirmará la eliminación.

Resultado esperado: La información registrada se almacenará en la base de datos, para los casos de ingreso y modificación. De aceptarse la eliminación, los datos serán eliminados de la base.

Evaluación de la prueba: Prueba satisfactoria.

CAPÍTULO III

3. MARCO DE RESULTADOS, DISCUSIÓN Y ANÁLISIS DE RESULTADOS

En este capítulo se analizó e interpretó los resultados de la aceptación del sistema mediante la técnica de observación, la cual nos permitió hacer un estudio estadístico sin la utilización del sistema y un análisis estadístico con la utilización del sistema en un ambiente prototipo.

Para este estudio nos enfocamos en la historia clínica por ser un proceso que está formado por gran parte de los datos importantes del historial de salud de un paciente, siendo para el médico esta información útil y necesaria para diagnosticar y dar el tratamiento apropiado a una persona que padece una determinada enfermedad o enfermedades.

En consideración a lo expuesto anteriormente nuestro estudio, discusión y análisis de resultados se centra en las historias clínicas de los pacientes del Hospital Básico San Marcos.

Seguidamente se procedió a obtener la muestra con la que se trabajó para realizar los análisis estadísticos correspondientes para cada uno de los procesos de inserción, actualización y consulta en la base de datos, la muestra se consiguió de acuerdo al número de flujo de pacientes tomados en un día de trabajo de un médico, en el mismo se observó que en promedio cada médico atiende a 10 paciente cada día, por este motivo se decidió tomar como muestra el valor de 10 para nuestro análisis.

3.1 Estudio de aceptación del Sistema

Para la aceptación del sistema como anteriormente ~~se explicó~~ se practicaron pruebas de observación las cuales se exponen más adelante con sus respectivos tablas y gráficos.

3.2 Diseño de experimentos

Para el diseño de experimentos se tomó en cuenta una comparación simple debido a que hay dos muestras, una con el sistema y otra sin el sistema, a continuación, se detallan los pasos a seguir.

Para el análisis estadístico la prueba que se utilizó es la t-Student para dos muestras independientes debido a que los datos son menores que 30, es decir, esta prueba trabaja con muestras pequeñas. Además, se consideró un nivel de significancia del 5% y un análisis a dos

colas, que se aplicó a los procesos de inserción, actualización y consulta de datos de las historias clínicas de los pacientes.

Seguidamente se consideró los siguientes planteamientos de hipótesis nula H_0 e hipótesis alternativa H_1 .

Planteamiento de la hipótesis nula

$$H_0: \mu_1 = \mu_2$$

Planteamiento de la hipótesis alternativa

$$H_1: \mu_1 \neq \mu_2$$

Donde μ_1 representa la media del tratamiento sin el sistema

Donde μ_2 representa la media del tratamiento con el sistema

Nivel de significancia: 5%

Numero de Observaciones: 10

3.3 Análisis estadístico en la inserción de datos

Se procedió a obtener la muestra para realizar los análisis estadísticos correspondientes para el proceso de inserción en la base de datos, la misma que se consiguió de acuerdo al número de flujo de pacientes tomados en un día de trabajo de un médico, en el mismo se observó que en promedio cada médico atiende a 10 paciente cada día, por este motivo se decidió tomar como muestra el valor de 10 para el análisis.

Tabla 1-3: Tiempo de respuesta en la inserción de datos

Inserción(segundos)		
Número de paciente	Sin el sistema	Con el sistema
1	240	60
2	180	35
3	200	31
4	160	60
5	220	55
6	160	41
7	140	43
8	260	38

9	240	52
10	220	60
Promedio	202	47,5

Realizado por: Fausto Orozco, 2017

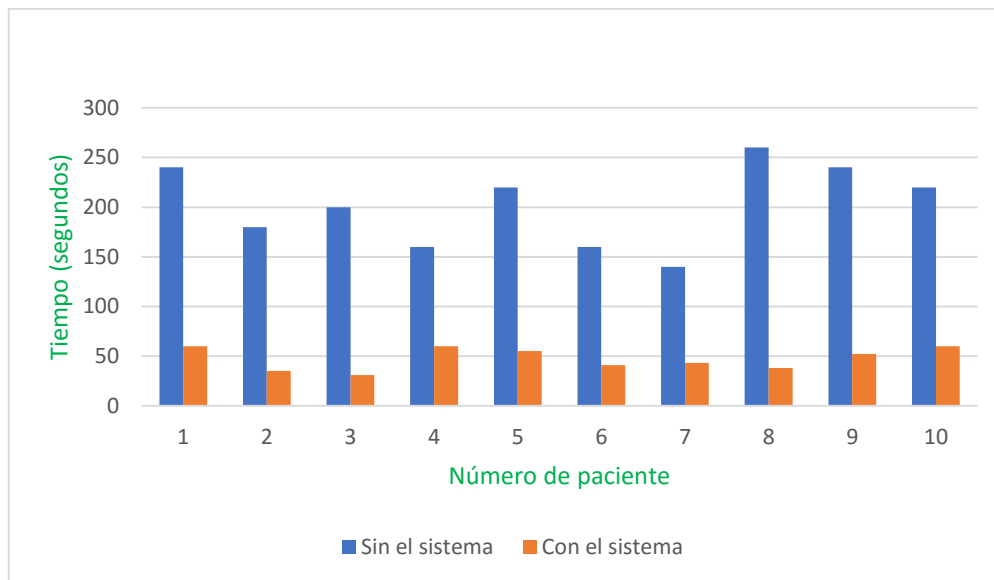


Gráfico 1-3 Tiempo de respuesta en la inserción de datos

Realizado por: Fausto Orozco, 2017

1. Redacción de la hipótesis

$H_1 = \textit{Existe}$ una diferencia significativa entre la media del tiempo sin el sistema para *ingresar los datos de la historia clínica del paciente* y la media del tiempo con el sistema para *ingresar los datos de la historia clínica del paciente*.

$H_0 = \textit{No Existe}$ una diferencia significativa entre la media del tiempo sin el sistema para *ingresar los datos de la historia clínica del paciente* y la media del tiempo con el sistema para *ingresar los datos de la historia clínica del paciente*.

2. Determinar el nivel significancia α

$$\alpha = 0,050$$

Es el porcentaje de error que se estuvo dispuesto a correr en la realización de la prueba estadística.

3. Elección de la prueba estadística

La prueba estadística que se utilizó es t-Student (muestras independientes) debido a que los datos de la muestra es 10 y considerando un análisis a dos colas.

4. Lectura de P-Valor

Para conocer si los datos que se obtuvieron siguieron una distribución normal en el proceso de inserción de datos, se hizo uso del software SPSS, este programa presentó los valores obtenidos para un conjunto de datos dado.

Para utilizar la distribución t-Student, fue fundamental que comprobemos, antes de empezar, si se daba el supuesto de normalidad para la inserción de datos (esto es, si se ajusta a una distribución normal la variable tiempo para cada uno de los grupos).

NORMALIDAD

Para el estudio de la normalidad de las dos poblaciones, sin el sistema y con el sistema, se analizó la prueba de Shapiro Wilk pues permitió trabajar con una muestra menor o igual a 30 observaciones, es decir, en las observaciones se consideró un valor de $n = 10$ al momento de hacer la inserción de datos. En la Tabla 2-3 se obtuvo las dos pruebas que permitieron comprobar si cumplen o no el supuesto de normalidad, esta fue obtenida desde el SPSS la cual exhibió sus resultados, siendo las pruebas que presentó el programa las siguientes:

- **Prueba de Kolmogorov-Smirnov** Permite trabajar con muestras grandes (>30 pacientes).
- **Prueba de Shapiro-Wilk** Permite trabajar con muestras pequeñas (≤ 30 pacientes).

Para ambas muestras tanto con el sistema y sin el sistema, se observó la salida de las pruebas de normalidad de *Shapiro-Wilk*, estas se analizaron más adelante:

Tabla 2-3: Prueba de normalidad de Shapiro-Wilk del tiempo de respuesta en la inserción de datos

Pruebas de normalidad							
Tiempo de Respuesta		Kolmogorov-Smirnov			Shapiro-Wilk		
		Estadístico	gl	Sig.	Estadístico	gl	Sig.
Valor de Tiempo de Respuesta	Sin el Sistema	,172	10	,200*	,944	10	,596
	Con el Sistema	,168	10	,200*	,890	10	,168

Realizado por: Fausto Orozco, 2017

Criterio para determinar Normalidad:

Se partió de las siguientes hipótesis estadísticas, asumiendo como nivel de significación $\alpha = 0,05$.

La hipótesis nula y alternativa para la variable cuantitativa, tiempo de respuesta, *sin* el sistema para la inserción de datos, se describe a continuación:

H_0 : La variable aleatoria tiempo *sin* el sistema tiene una distribución normal.

H_1 : La variable aleatoria tiempo *sin* el sistema *no* tiene una distribución normal.

La hipótesis nula y alternativa para la variable cuantitativa, tiempo de respuesta, *con* el sistema para la inserción de datos, se describe a continuación:

H_0 : La variable aleatoria tiempo *con* el sistema tiene una distribución normal.

H_1 : La variable aleatoria tiempo *con* el sistema *no* tiene una distribución normal.

Para determinar si se acepta o se rechaza cada una de las hipótesis planteadas se aplicaron los siguientes apartados:

$P\text{-Valor} > \alpha$ Aceptar H_0 = Los datos provienen de una distribución **normal**.

$P\text{-Valor} < \alpha$ Aceptar H_1 = Los datos **NO** provienen de una distribución **normal**.

A continuación, se procedió a concluir si ambas muestras, tanto con el sistema y sin el sistema, cumplen con el supuesto de normalidad, como valor de probabilidad calculado se escogió el nivel de significancia de *Shapiro-Wilk*, ver Tabla 2-3, para el caso sin el sistema es 0,596 y lo comparamos con el nivel de significancia 0,050, y para el caso con el sistema es 0,168 igualmente lo comparamos con el nivel de significancia 0,050, ver Tabla 3-3.

Tabla 3-3: Decisión estadística de la normalidad de Shapiro-Wilk en la inserción de datos

NORMALIDAD Tiempos		
Valor de Probabilidad Obtenido	Relación	Nivel de Significancia
P-Valor (<i>Sin el sistema</i>) = 0,596	>	$\alpha = 0,050$
P-Valor (<i>Con el sistema</i>) = 0,168	>	$\alpha = 0,050$
CONCLUSIÓN: La variable aleatoria tiempo en ambos grupos se comporta normalmente, es decir, se acepta la hipótesis nula de normalidad H_0 de la distribución en cada caso (sin el sistema y con el sistema).		

Realizado por: Fausto Orozco, 2017

Se concluye que ambas distribuciones de datos, es decir, la muestra del tiempo sin el sistema y la muestra del tiempo con el sistema para la inserción de datos, proceden de una distribución normal, por esta razón, se acepta la hipótesis nula H_0 y se rechaza la hipótesis

alternativa H_1 . Por consiguiente, se procedió a realizar la prueba de Levene para verificar la igualdad de varianzas.

IGUALDAD DE VARIANZA

La prueba de Levene para la igualdad de varianzas nos indica si podemos o no suponer varianzas iguales, esto nos permitirá, conocer en qué caso nos encontramos, para poder hacer la lectura de los resultados en la fila “Se asumen varianzas iguales” o “No se asumen varianzas iguales” en la Tabla 4-3. De hecho, SPSS lo realiza automáticamente siempre que se desea realizar la prueba t-Student, y muestra los resultados para la prueba t-Student, tanto en el caso de que se desee admitir o rechazar la igualdad de varianzas.

Se partió de las hipótesis estadísticas para contrastar el supuesto de homocedasticidad u homogeneidad de la varianza, tomando como nivel de significación $\alpha = 0,05$.

Se describió, la hipótesis nula y alternativa para la igualdad de varianzas, *sin* el sistema y con el sistema, para la inserción de datos:

H_0 : Las varianzas sin el sistema y con el sistema para la inserción de datos ***son iguales***.

H_1 : Las varianzas sin el sistema y con el sistema para la inserción de datos ***no son iguales***.

Se procedió a hacer un análisis en las dos hipótesis planteadas, para esto se observó la fila de “Se asumen varianzas iguales” y “No se asumen varianzas iguales”, ver Tabla 4-3, los resultados se observan en la salida de la prueba de *Levene* para la igualdad de varianzas, estas se muestran a continuación:

Tabla 4-3: Prueba de Levene y Prueba t-Student en la inserción de datos

Prueba de muestras independientes								
	Prueba de Levene para la igualdad de varianzas		Prueba t para la igualdad de medias					
	F	Sig.	t	gl	Sig. (bilateral)	Diferencia de medias	Diferencia de error estándar	95% de intervalo de confianza de la diferencia
								Inferior Superior

Valor de Tiempo de Respuesta	Se asumen varianzas iguales	15,62	.001	11,63	18	.000	154,50	13,29	126,59	182,41
	No se asumen varianzas iguales			11,63	10,37	.000	154,50	13,29	125,04	183,96

Realizado por: Fausto Orozco, 2017

Criterio para determinar Igualdad de Varianza:

Para determinar si las varianzas son iguales o no a través de las hipótesis planteadas se aplicaron los siguientes apartados:

$P\text{-Valor} > \alpha$ Aceptar H_0 = Las varianzas son iguales.

$P\text{-Valor} < \alpha$ Aceptar H_1 = Existe diferencia significativa entre las varianzas.

A continuación, se procedió a concluir si se cumple o no con el supuesto de igualdad de varianzas, como valor de probabilidad calculado se seleccionó el valor 0,000, que es el nivel de significancia de *Levene* (columna “Prueba de Levene para la igualdad de varianzas”), de la Tabla 4-3, para compararlo con el nivel de significancia $\alpha = 0,050$, ver Tabla 5-3.

Tabla 5-3: Decisión estadística de la igualdad de varianzas en la inserción de datos

IGUALDAD DE VARIANZA		
Valor de Probabilidad Obtenido	Relación	Nivel de Significancia
P-Valor = 0,000	<	$\alpha = 0,050$
CONCLUSIÓN:		
La varianza muestral sin el sistema y con el sistema tienen una diferencia estadísticamente significativa, es decir, se rechaza la hipótesis nula de igualdad de varianzas H_0 y se acepta la hipótesis alternativa H_1 .		

Realizado por: Fausto Orozco, 2017

Se concluyó entonces que las dos varianzas sin el sistema y con el sistema para la inserción de datos tenían una diferencia estadísticamente significativa, por esta razón, se rechazó la hipótesis nula H_0 y se aceptó la hipótesis alternativa H_1 . De tal manera que, se pudo realizar la prueba t de Student de comparación de dos medias, grupos o muestras independientes de carácter paramétrico.

5. Prueba t-Student para la igualdad de medias

Para el análisis de la comparación de medias de las dos variables cuantitativas tiempo, sin el sistema y con el sistema para la inserción de datos, se estudiará la prueba de t-Student.

Después de comprobar la normalidad y la igualdad de varianza de las variables tiempo. Finalmente se ha llegado al punto, en la que ya se puede dar una conclusión sobre si las medias del tiempo, con el sistema y sin el sistema para la inserción de datos, son iguales o existe una diferencia estadísticamente significativa entre las medias de las dos muestras.

Las hipótesis nula y alternativa para la igualdad de medias del tiempo, *sin* el sistema y con el sistema, para la inserción de datos quedaron formalmente enunciadas de la siguiente manera:

H_0 : Las medias del tiempo sin el sistema y con el sistema para la inserción de datos ***son iguales***.

H_1 : Las medias del tiempo sin el sistema y con el sistema para la inserción de datos ***no son iguales***.

Criterio para determinar la igualdad de medias:

Si la probabilidad obtenida P-Valor $> \alpha$, no rechace H_0 (Se acepta H_0).

Si la probabilidad obtenida P-Valor $< \alpha$, rechace H_0 (Se acepta H_1).

Inmediatamente, se procedió a concluir si se cumple con la hipótesis nula de igualdad de medias del tiempo H_0 de la prueba t-Student, como valor de probabilidad calculado se seleccionó el resultado 0,000, que es el valor del estadístico t de la distribución t-Student (columna “Prueba t para la igualdad de medias”), ver Tabla 4-3, y a su vez este valor será comparado con el valor de significancia $\alpha = 0,050$ previamente asignado, ver Tabla 6-3.

Tabla 6-3: Decisión estadística de la prueba t-Student para la igualdad de medias del tiempo en la inserción de datos

Prueba t-Student para la igualdad de medias del tiempo		
Valor de Probabilidad Obtenido	Relación	Nivel de Significancia
P-Valor = 0,000	$<$	$\alpha = 0,050$
CONCLUSIÓN: <i>Existe una diferencia estadísticamente significativa</i> entre el tiempo medio sin el sistema y el tiempo medio con el sistema para la inserción de datos, es decir, se rechaza la hipótesis nula de igualdad de medias H_0 y se acepta la hipótesis alternativa H_1 .		

Realizado por: Fausto Orozco, 2017

Según el análisis estadístico de la prueba t-Student se observó que existe una diferencia estadísticamente significativa entre la media del tiempo sin el sistema y la media del tiempo con el sistema para la inserción de datos de las historias clínicas de los pacientes, por lo tanto, se rechazó la hipótesis nula H_0 y se aceptó la hipótesis alternativa H_1 .

Haciendo uso de la herramienta de office, Excel 2016, los valores de la siguiente tabla se obtuvieron de aplicar la prueba t de student para dos muestras suponiendo varianzas desiguales.

Tabla 7-3: Datos estadísticos t-Student en la inserción de datos

Inserción de Datos		
Prueba t para dos muestras suponiendo varianzas desiguales		
	<i>Sin el sistema</i>	<i>Con el sistema</i>
Media	202	47,5
Varianza	1640	125,167
Observaciones	10	10
Diferencia hipotética de las medias	0	
Grados de libertad	10	
<u>Estadístico t</u>	11,62881938	
<u>P(T<=t) una cola</u>	0,000000196	
<u>Valor crítico de t (una cola)</u>	1,812461123	
<u>P(T<=t) dos colas</u>	0,000000392	
<u>Valor crítico de t (dos colas)</u>	2,228138852	

Realizado por: Fausto Orozco, 2017

Tiempos:

Para un análisis comparativo en la mejora de los tiempos de respuesta con el empleo del sistema para el proceso de inserción de los datos de las historias clínicas de los pacientes, se aplicó el método matemático que permite obtener el factor de proporcionalidad del cuarto término desconocido a partir de los valores de los otros tres términos conocidos.

Esto es, se trata de una operación que nos ayuda a resolver rápidamente problemas de *proporcionalidad*, es decir, para este proceso necesitamos 3 datos: dos magnitudes proporcionales entre sí, la primera es el *tiempo medio total sin el sistema* con un valor de 202 segundos y la segunda es el *porcentaje del tiempo medio total sin el sistema* con un valor de 100 %, y una tercera magnitud que es el *tiempo medio total con el sistema* con un valor de 47,50 segundos. A partir de estos tres datos conocidos, averiguaremos el cuarto término desconocido de la proporcionalidad.

En consideración se puede observar el cálculo del cuarto término de la proporcionalidad, obteniendo de esta manera el nuevo *porcentaje del tiempo medio total con el sistema*, su resultado fue de 23,51 %.

$$\begin{array}{cc} 202 & 100 \% \\ 47,5 & X \end{array}$$

$$X = \frac{47,5 \times 100\%}{202} = 23,51 \%$$

Siendo el 100 % el porcentaje del tiempo medio total del proceso realizado manualmente sin el sistema, se procedió a realizar la diferencia con el 23,51 % que es el nuevo valor de porcentaje del tiempo medio total del proceso realizado automáticamente con el sistema, de donde se obtuvo el valor de 76,49 %, esto es:

$$100\% - 23,51\% = 76,49\%$$

Se concluyó por lo tanto que, en el proceso de inserción de los datos de las historias clínicas de los pacientes con el sistema, se ha mejorado el tiempo promedio en un valor del 76,49 %.

A continuación, se presenta un gráfico de barras en el que se observa el tiempo medio sin el sistema y el tiempo medio con el sistema, para la inserción de los datos de las historias clínicas de los pacientes.

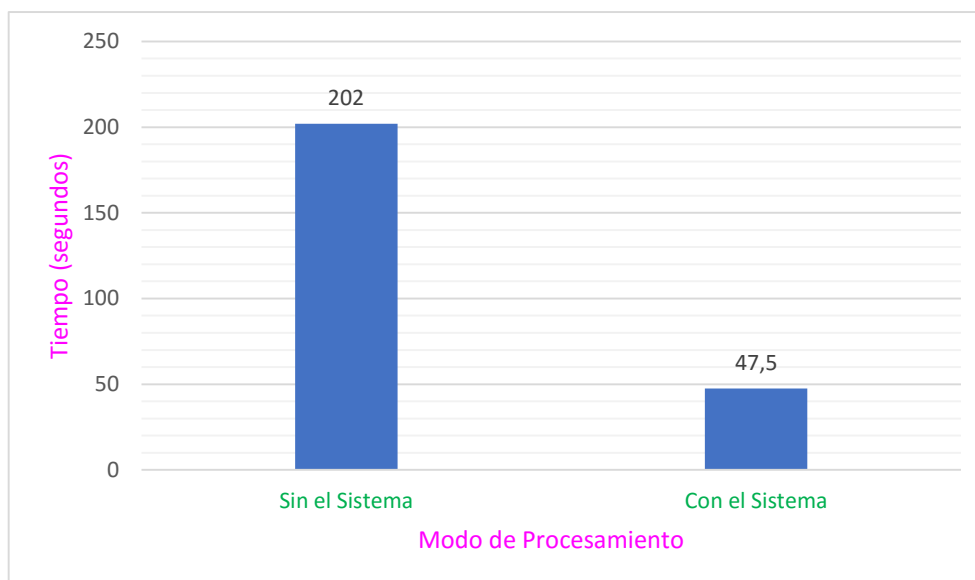


Gráfico 2-3 Tiempo de respuesta medio en la inserción de datos

Realizado por: Fausto Orozco, 2017

3.4 Análisis estadístico en la actualización de datos

Seguidamente se procedió a obtener la muestra para realizar los análisis estadísticos correspondientes para el proceso de actualización en la base de datos, la muestra se consiguió de acuerdo al número de flujo de pacientes tomados en un día de trabajo de un médico, en el mismo se observó que en promedio cada médico atiende a 10 paciente cada día, por este motivo se decidió tomar a 10 como el valor de la muestra para el análisis.

Tabla 8-3: Tiempo de respuesta en la actualización de datos

Actualización(segundos)		
Número de paciente	Sin el sistema	Con el sistema
1	160	25
2	100	19
3	100	32
4	120	27
5	160	31
6	100	40
7	140	23
8	80	30
9	180	26
10	140	28
Promedio	128	28,1

Realizado por: Fausto Orozco, 2017

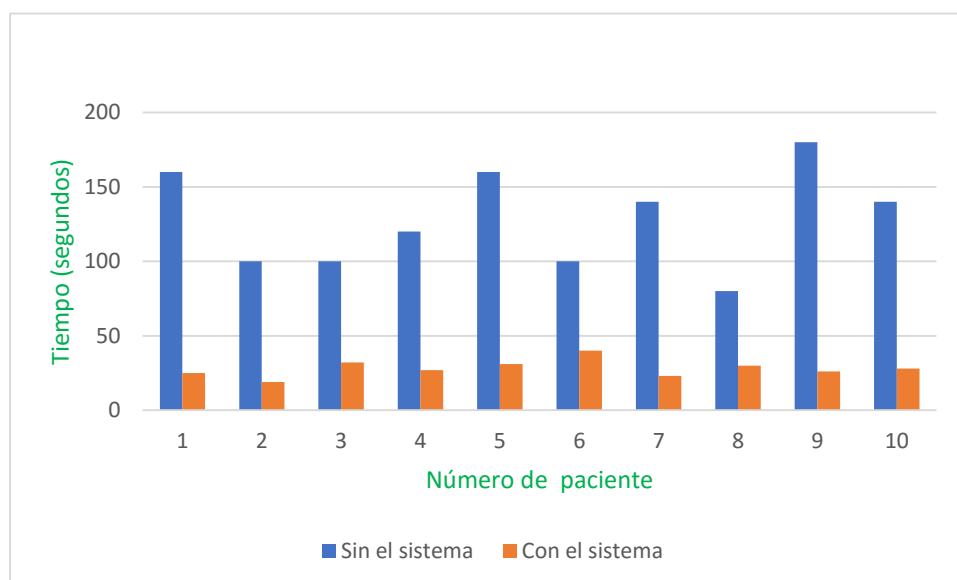


Gráfico 3-3 Tiempo de respuesta en la actualización de dato

Realizado por: Fausto Orozco, 2017

1. Redacción de la hipótesis

$H_1 = \text{Existe}$ una diferencia significativa entre la media del tiempo sin el sistema para actualizar los datos de la historia clínica del paciente y la media del tiempo con el sistema para actualizar los datos de la historia clínica del paciente.

$H_0 = \text{No Existe}$ una diferencia significativa entre la media del tiempo sin el sistema para actualizar los datos de la historia clínica del paciente y la media del tiempo con el sistema para actualizar los datos de la historia clínica del paciente.

2. Determinar el nivel significancia α

$$\alpha = 0,050$$

Es el porcentaje de error que se estuvo dispuesto a correr en la realización de la prueba estadística.

3. Elección de la prueba estadística

La prueba estadística que se utilizó es t-Student (muestras independientes) debido a que los datos de la muestra es 10 y considerando un análisis a dos colas.

4. Lectura de P-Valor

Para el proceso de la actualización de datos se realizó el mismo estudio para conocer si los datos que se obtuvieron siguieron una distribución normal, se hizo uso del software SPSS, el mismo que presentó los valores obtenidos para su respectivo análisis.

Para utilizar la distribución t-Student, fue fundamental que comprobemos, antes de empezar, si se daba el supuesto de normalidad para la actualización de datos (esto es, si se ajusta a una distribución normal la variable tiempo para cada uno de los grupos).

NORMALIDAD

Para el estudio de la normalidad de las dos poblaciones, sin el sistema y con el sistema, se analizó la prueba de Shapiro Wilk ya que es ésta la que permitió trabajar con una muestra menor o igual a 30 observaciones, es decir, se consideró un valor de $n = 10$ observaciones para la actualización de datos. En la Tabla 9-3 se obtuvo las dos pruebas que permitieron comprobar si cumplen o no el supuesto de normalidad, esta fue obtenida desde el SPSS la cual exhibió sus resultados, siendo las pruebas que se presentaron las siguientes:

- **Prueba de Kolmogorov-Smirnov** Permite trabajar con muestras grandes (>30 pacientes).
- **Prueba de Shapiro-Wilk** Permite trabajar con muestras pequeñas (<=30 pacientes).

Para ambas muestras tanto con el sistema y sin el sistema, se observó la salida de las pruebas de normalidad de *Shapiro-Wilk*, la cual es de nuestro interés:

Tabla 9-3: Prueba de normalidad de Shapiro-Wilk del tiempo de respuesta en la actualización de datos

Pruebas de normalidad							
Tiempo de Respuesta		Kolmogorov-Smirnov			Shapiro-Wilk		
		Estadístico	gl	Sig.	Estadístico	gl	Sig.
Valor de Tiempo de Respuesta	Sin el Sistema	,202	10	,200*	,938	10	,532
	Con el Sistema	,147	10	,200*	,967	10	,858

Realizado por: Fausto Orozco, 2017

Criterio para determinar Normalidad:

Las hipótesis estadísticas que se consideraron son las siguientes, asumiendo como nivel de significación $\alpha = 0,05$.

La hipótesis nula y alternativa para la variable cuantitativa, tiempo de respuesta para la actualización de datos, *sin* el sistema, se representa a continuación:

H_0 : La variable aleatoria tiempo *sin* el sistema tiene una distribución normal.

H_1 : La variable aleatoria tiempo *sin* el sistema *no* tiene una distribución normal.

La hipótesis nula y Alternativa para la variable cuantitativa, tiempo de respuesta para la actualización de datos, *con* el sistema, se representa a continuación:

H_0 : La variable aleatoria tiempo *con* el sistema tiene una distribución normal.

H_1 : La variable aleatoria tiempo *con* el sistema *no* tiene una distribución normal.

Para determinar si se acepta o se rechaza cada una de las hipótesis planteadas se aplicaron los siguientes apartados:

$P\text{-Valor} > \alpha$ Aceptar $H_0 =$ Los datos provienen de una distribución **normal**.

$P\text{-Valor} < \alpha$ Aceptar $H_1 =$ Los datos **NO** provienen de una distribución **normal**.

A continuación, se procedió a concluir si ambas muestras, tanto con el sistema y sin el sistema, cumplen con el supuesto de normalidad, como valor de probabilidad calculado se escogió el nivel de significancia de *Shapiro-Wilk*, ver Tabla 9-3, para el caso sin el sistema es 0,532 y lo comparamos con el nivel de significancia 0,050, y para el caso con el sistema es 0,858 igualmente lo comparamos con el nivel de significancia 0,050, ver Tabla 10-3.

Tabla 10-3: Decisión estadística de la normalidad de Shapiro-Wilk en la actualización de datos

NORMALIDAD <i>Tiempos</i>		
Valor de Probabilidad Obtenido	Relación	Nivel de Significancia
P-Valor (<i>Sin el sistema</i>) = 0,532	>	$\alpha = 0,050$
P-Valor (<i>Con el sistema</i>) = 0,858	>	$\alpha = 0,050$
CONCLUSIÓN: La variable aleatoria tiempo en ambos grupos se comporta normalmente, es decir, se acepta la hipótesis nula de normalidad H_0 de la distribución en cada caso (sin el sistema y con el sistema).		

Realizado por: Fausto Orozco, 2017

Se concluyó que ambas distribuciones de datos, es decir, la muestra del tiempo sin el sistema y la muestra del tiempo con el sistema para la actualización de datos, proceden de una distribución normal, por esta razón, se acepta la hipótesis nula H_0 y se rechaza la hipótesis alternativa H_1 . Por consiguiente, se procedió a realizar la prueba de Levene para verificar la igualdad de varianzas.

IGUALDAD DE VARIANZA

La prueba de Levene para la igualdad de varianzas nos indica si podemos o no suponer varianzas iguales, esto permitirá, identificar para el proceso de actualización de datos, en qué caso nos encontramos para hacer la lectura de los resultados en la fila “Se asumen varianzas iguales” o “No se asumen varianzas iguales” en la Tabla 11-3. De hecho, SPSS lo realiza automáticamente siempre que se desea realizar la prueba t-Student, y muestra los resultados, tanto en el caso de que se desee admitir o rechazar la igualdad de varianzas.

Se partió de las hipótesis estadísticas para contrastar el supuesto de homocedasticidad u homogeneidad de la varianza, tomando como nivel de significación $\alpha = 0,05$.

Se describió, la hipótesis nula y alternativa para la igualdad de varianzas, *sin* el sistema y con el sistema, para la actualización de datos:

H_0 : Las varianzas sin el sistema y con el sistema para la actualización de datos *son iguales*.

H_1 : Las varianzas sin el sistema y con el sistema para la actualización de datos **no son iguales**.

Considerando las dos hipótesis planteadas se procedió a realizar un análisis, para esto se observó en la fila que se tiene con encabezado “Se asumen varianzas iguales” o “No se asumen varianzas iguales”, ver Tabla 11-3, los resultados se observaron en la salida de la prueba de *Levene* para la igualdad de varianzas, estas se muestran a continuación:

Tabla 11-3: Prueba de Levene y Prueba t-Student en la actualización de datos

Prueba de muestras independientes										
		Prueba de Levene para la igualdad de varianzas		Prueba t para la igualdad de medias						
		F	Sig.	t	gl	Sig. (bilateral)	Diferencia de medias	Diferencia de error estándar	95% de intervalo de confianza de la diferencia	
									Inferior	Superior
Valor de Tiempo de Respuesta	Se asumen varianzas iguales	25,12	.000	9,45	18	.000	99,90	10,57	77,70	122,10
	No se asumen varianzas iguales			9,45	9,54	.000	99,90	10,57	76,20	123,60

Realizado por: Fausto Orozco, 2017

Criterio para determinar Igualdad de Varianza:

Para determinar si las varianzas son iguales o no a través de las hipótesis planteadas se aplicaron los siguientes apartados:

$P\text{-Valor} > \alpha$ Aceptar H_0 = Las varianzas son iguales.

$P\text{-Valor} < \alpha$ Aceptar H_1 = Existe diferencia significativa entre las varianzas.

En seguida, se procedió a concluir si se cumple o no con el supuesto de igualdad de varianzas, como valor de probabilidad calculado se seleccionó el valor 0,000, que es el nivel de significancia de *Levene* (columna “Prueba de Levene para la igualdad de varianzas”), de la Tabla 11-3, para compararlo con el nivel si significancia $\alpha = 0,050$, ver Tabla 12-3.

Tabla 12-3: Decisión estadística de la igualdad de varianzas en la actualización de datos

IGUALDAD DE VARIANZA		
Valor de Probabilidad Obtenido	Relación	Nivel de Significancia
P-Valor = 0,000	<	$\alpha = 0,050$
CONCLUSIÓN: La varianza muestral sin el sistema y con el sistema tienen una diferencia estadísticamente significativa, es decir, se rechaza la hipótesis nula de igualdad de varianzas H_0 y se acepta la hipótesis alternativa H_1 .		

Realizado por: Fausto Orozco, 2017

Se concluyó entonces que las dos varianzas sin el sistema y con el sistema para la actualización de datos tenían una diferencia estadísticamente significativa, por consiguiente, se rechazó la hipótesis nula H_0 y se aceptó la hipótesis alternativa H_1 . En consecuencia, se pudo realizar la prueba t de Student de comparación de dos medias, grupos o muestras independientes de carácter paramétrico.

5. Prueba t-Student para la igualdad de medias

Para el análisis de la comparación de medias de las dos variables cuantitativas tiempo, sin el sistema y con el sistema para la actualización de datos, se tomará la prueba de t-Student.

Después de comprobar la normalidad y la igualdad de varianza de las variables tiempo. Por último, se ha llegado al punto, en la que ya se puede dar una conclusión sobre si las medias del tiempo, con el sistema y sin el sistema para la actualización de datos, son iguales o existe una diferencia estadísticamente significativa entre las medias de las dos muestras.

Las hipótesis nula y alternativa para la igualdad de medias del tiempo, *sin* el sistema y con el sistema, para la actualización de datos quedaron formalmente enunciadas de la siguiente manera:

H_0 : Las medias del tiempo sin el sistema y con el sistema para la actualización de datos ***son iguales***.

H_1 : Las medias del tiempo sin el sistema y con el sistema para la actualización de datos ***no son iguales***.

Criterio para determinar la igualdad de medias:

Si la probabilidad obtenida P-Valor $> \alpha$, no rechace H_0 (Se acepta H_0).

Si la probabilidad obtenida P-Valor $< \alpha$, rechace H_0 (Se acepta H_1).

Seguidamente, se procedió a concluir si se cumple con la hipótesis nula de igualdad de medias del tiempo H_0 de la prueba t-Student, como valor de probabilidad calculado se escogió el resultado 0,000, que es el valor del estadístico t de la distribución t-Student (columna “Prueba t para la igualdad de medias”), ver Tabla 11-3, y a su vez este valor será comparado con el valor de significancia $\alpha = 0,050$ previamente asignado, ver Tabla 13-3.

Tabla 13-3: Decisión estadística de la prueba t-Student para la igualdad de medias del tiempo en la actualización de datos

Prueba t-Student para la igualdad de medias del tiempo		
Valor de Probabilidad Obtenido	Relación	Nivel de Significancia
P-Valor = 0,000	<	$\alpha = 0,050$
CONCLUSIÓN: <i>Existe una diferencia estadísticamente significativa</i> entre el tiempo medio sin el sistema y el tiempo medio con el sistema para la actualización de datos, es decir, se rechaza la hipótesis nula de igualdad de medias H_0 y se acepta la hipótesis alternativa H_1 .		

Realizado por: Fausto Orozco, 2017

Según el análisis estadístico de la prueba t-Student se observó que existe una diferencia estadísticamente significativa entre la media del tiempo sin el sistema y la media del tiempo con el sistema para la actualización de datos de las historias clínicas de los pacientes, por consiguiente, se rechazó la hipótesis nula H_0 y se aceptó la hipótesis alternativa H_1 .

Haciendo uso de la herramienta de office, Excel 2016, los valores de la siguiente tabla se obtuvieron de aplicar la prueba t de student para dos muestras suponiendo varianzas desiguales.

Tabla 14-3: Datos estadísticos t-Student en la actualización de datos

Actualización de Datos		
Prueba t para dos muestras suponiendo varianzas desiguales		
	<i>Sin el sistema</i>	<i>Con el sistema</i>
Media	128	28,1
Varianza	1084,444444	32,5444444
Observaciones	10	10
Diferencia hipotética de las medias	0	
Grados de libertad	10	
<u>Estadístico t</u>	9,452377586	
P(T<=t) una cola	0,000001328	
<u>Valor crítico de t (una cola)</u>	1,812461123	

P(T≤t) dos colas	0,000002657	
Valor crítico de t (dos colas)	2,228138852	

Realizado por: Fausto Orozco, 2017

Tiempos:

Para un análisis comparativo en la mejora de los tiempos de respuesta con el empleo del sistema para el proceso de actualización de los datos de las historias clínicas de los pacientes, se aplicó el método matemático que permite obtener el factor de proporcionalidad del cuarto término desconocido a partir de los valores de los otros tres términos conocidos.

Esto es, se trata de una operación que nos ayuda a resolver rápidamente problemas de *proporcionalidad*, es decir, para este proceso necesitamos 3 datos: dos magnitudes proporcionales entre sí, la primera es el *tiempo medio total sin el sistema* con un valor de 128 segundos y la segunda es el *porcentaje del tiempo medio total sin el sistema* con un valor de 100 %, y una tercera magnitud que es el *tiempo medio total con el sistema* con un valor de 28,10 segundos. A partir de estos tres datos conocidos, averiguaremos el cuarto término desconocido de la proporcionalidad.

En consideración se puede observar el cálculo del cuarto término de la proporcionalidad, obteniendo de esta manera el nuevo *porcentaje del tiempo medio total con el sistema*, su resultado fue de 21,95 %.

$$\begin{array}{cc}
 128 & 100 \% \\
 28,1 & X \\
 \\
 X = \frac{28,1 \times 100\%}{128} = 21,95 \%
 \end{array}$$

Siendo el 100 % el porcentaje del tiempo medio total del proceso realizado manualmente sin el sistema, se procedió a realizar la diferencia con el 21,95 % que es el nuevo valor de porcentaje del tiempo medio total del proceso realizado automáticamente con el sistema, de donde se obtuvo el valor de 78,05 %, esto es:

$$100\% - 21,95\% = 78,05\%$$

Se concluyó por lo tanto que, en el proceso de actualización de los datos de las de las historias clínicas de los pacientes con el sistema, se ha mejorado el tiempo promedio en un valor del 78,05 %.

A continuación, se presenta un gráfico de barras en el que se observa el tiempo medio sin el sistema y el tiempo medio con el sistema, para la actualización de los datos de las historias clínicas de los pacientes.

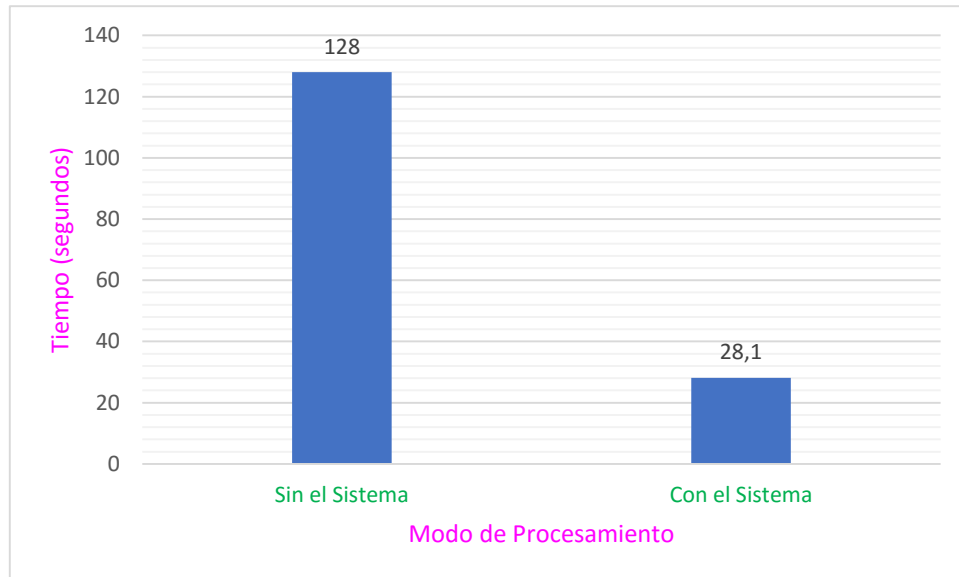


Gráfico 4-3 Tiempo de respuesta medio en la actualización de datos

Realizado por: Fausto Orozco, 2017

3.5 Análisis estadístico en la consulta de datos

Posteriormente se procedió a obtener la muestra para realizar los análisis estadísticos correspondientes para el proceso de consulta en la base de datos, la muestra se consiguió de acuerdo al número de flujo de pacientes tomados en un día de trabajo de un médico, en el mismo se observó que en promedio cada médico atiende a 10 paciente cada día, por este motivo se decidió tomar una muestra con un valor igual a 10 para el análisis.

Tabla 15-3: Tiempo de respuesta en la consulta de datos

Consulta(segundos)		
Número de paciente	Sin el sistema	Con el sistema
1	60	12
2	43,2	10
3	76,8	11,5
4	32,4	8,5
5	60	12
6	45	11
7	54,6	9,5
8	40,5	10,5

9	30	8
10	66	10,5
Promedio	50,85	10,35

Realizado por: Fausto Orozco, 2017

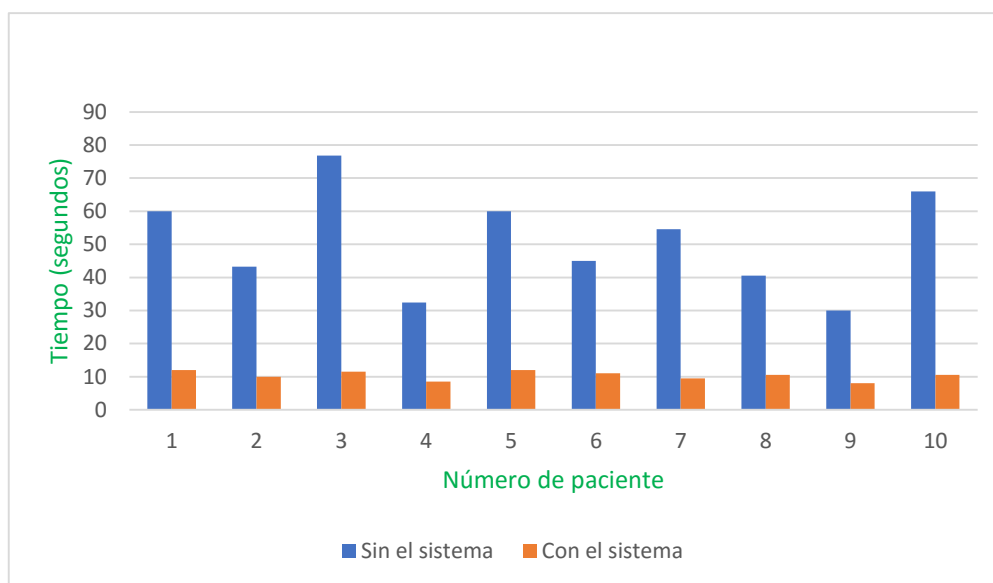


Gráfico 5-3 Tiempo de respuesta en la consulta de datos

1. Redacción de la hipótesis

$H_1 = \text{Existe}$ una diferencia significativa entre la media del tiempo sin el sistema para consultar los datos de la historia clínica del paciente y la media del tiempo con el sistema para consultar los datos de la historia clínica del paciente.

$H_0 = \text{No Existe}$ una diferencia significativa entre la media del tiempo sin el sistema para consultar los datos de la historia clínica del paciente y la media del tiempo con el sistema para consultar los datos de la historia clínica del paciente.

2. Determinar el nivel significancia α

$$\alpha = 0,050$$

Es el porcentaje de error que se estuvo dispuesto a correr en la realización de la prueba estadística.

3. Elección de la prueba estadística

La prueba estadística que se utilizó es t-Student (muestras independientes) debido a que los datos de la muestra es 10 y considerando un análisis a dos colas.

4. Lectura de P-Valor

Para el proceso de la consulta de datos se realizó el mismo estudio para conocer si los datos que se obtuvieron siguieron una distribución normal, se hizo uso del software SPSS, el mismo que presentó los valores obtenidos para su respectivo análisis.

Para hacer uso de la distribución t-Student, fue fundamental que comprobemos, antes de empezar, si se daba el supuesto de normalidad para la consulta de datos (esto es, si se ajusta a una distribución normal la variable tiempo para cada uno de los grupos).

NORMALIDAD

Para el estudio de la normalidad de las dos poblaciones, sin el sistema y con el sistema, se analizó la prueba de Shapiro Wilk ya que es ésta la que permitió trabajar con una muestra menor o igual a 30 observaciones, es decir, para la consulta de datos se consideró un valor de $n = 10$ observaciones. En la Tabla 16-3 se obtuvo las dos pruebas que permitieron comprobar si cumplen o no el supuesto de normalidad, esta fue obtenida desde el SPSS la cual exhibió sus resultados, y las pruebas presentadas son las siguientes:

- **Prueba de Kolmogorov-Smirnov** Permite trabajar con muestras grandes (>30 pacientes).
- **Prueba de Shapiro-Wilk** Permite trabajar con muestras pequeñas (≤ 30 pacientes).

Para ambas muestras tanto con el sistema y sin el sistema, se observó la salida de las pruebas de normalidad de *Shapiro-Wilk*, las mismas que se analizó:

Tabla 16-3: Prueba de normalidad de Shapiro-Wilk del tiempo de respuesta en la consulta de datos

Pruebas de normalidad							
Tiempo de Respuesta		Kolmogorov-Smirnov			Shapiro-Wilk		
		Estadístico	gl	Sig.	Estadístico	gl	Sig.
Valor de Tiempo de Respuesta	Sin el Sistema	,151	10	,200*	,963	10	,818
	Con el Sistema	,143	10	,200*	,939	10	,539

Realizado por: Fausto Orozco, 2017

Criterio para determinar Normalidad:

Asumiendo como nivel de significación $\alpha = 0,05$, se consideraron las siguientes hipótesis estadísticas.

La hipótesis nula y alternativa para la variable cuantitativa, tiempo de respuesta para la consulta de datos, *sin* el sistema, se representa a continuación:

H_0 : La variable aleatoria tiempo *sin* el sistema tiene una distribución normal.

H_1 : La variable aleatoria tiempo *sin* el sistema *no* tiene una distribución normal.

La hipótesis nula y alternativa para la variable cuantitativa, tiempo de respuesta para la consulta de datos, *con* el sistema, se representa a continuación:

H_0 : La variable aleatoria tiempo *con* el sistema tiene una distribución normal.

H_1 : La variable aleatoria tiempo *con* el sistema *no* tiene una distribución normal.

Para las hipótesis planteadas se aplicaron los siguientes apartados, para determinar si se acepta o se rechaza cada una estas:

$P\text{-Valor} > \alpha$ Aceptar $H_0 =$ Los datos provienen de una distribución **normal**.

$P\text{-Valor} < \alpha$ Aceptar $H_1 =$ Los datos **NO** provienen de una distribución **normal**.

Inmediatamente, se procedió a concluir si ambas muestras, tanto con el sistema y sin el sistema, cumplen con el supuesto de normalidad, como valor de probabilidad calculado se escogió el nivel de significancia de *Shapiro-Wilk*, ver Tabla 16-3, para el caso sin el sistema es 0,818 y lo comparamos con el nivel de significancia 0,050, y para el caso con el sistema es 0,539 igualmente lo comparamos con el nivel de significancia 0,050, ver Tabla 17-3.

Tabla 17-3: Decisión estadística de la normalidad de Shapiro-Wilk en la consulta de datos

NORMALIDAD Tiempos		
Valor de Probabilidad Obtenido	Relación	Nivel de Significancia
P-Valor (<i>Sin el sistema</i>) = 0,818	>	$\alpha = 0,050$
P-Valor (<i>Con el sistema</i>) = 0,539	>	$\alpha = 0,050$
CONCLUSIÓN: La variable aleatoria tiempo en ambos grupos se comporta normalmente, es decir, se acepta la hipótesis nula de normalidad H_0 de la distribución en cada caso (sin el sistema y con el sistema).		

Realizado por: Fausto Orozco, 2017

Se concluyó que ambas distribuciones de datos, es decir, la muestra del tiempo sin el sistema y la muestra del tiempo con el sistema para la consulta de datos, proceden de una distribución normal, por esta razón, se acepta la hipótesis nula H_0 y se rechaza la hipótesis alternativa H_1 .

De manera que, se procedió a realizar la prueba de Levene para verificar la igualdad de varianzas.

IGUALDAD DE VARIANZA

La prueba de Levene para la igualdad de varianzas nos indica si podemos o no suponer varianzas iguales, esto nos permitirá, conocer en qué caso nos encontramos, para hacer la lectura de los resultados en la fila “Se asumen varianzas iguales” o “No se asumen varianzas iguales” en la Tabla 18-3. De hecho, SPSS lo realiza automáticamente siempre que se desea realizar la prueba t-Student, y muestra los resultados para la prueba t-Student, tanto en el caso de que se desee admitir o rechazar la igualdad de varianzas.

Se partió de las hipótesis estadísticas para contrastar el supuesto de homocedasticidad u homogeneidad de la varianza, tomando como nivel de significación $\alpha = 0,05$.

Se describió, la hipótesis nula y alternativa para la igualdad de varianzas, *sin* el sistema y con el sistema, para la consulta de datos:

H_0 : Las varianzas sin el sistema y con el sistema para la consulta de datos *son iguales*.

H_1 : Las varianzas sin el sistema y con el sistema para la consulta de datos *no son iguales*.

Se procedió a hacer un análisis en las dos hipótesis planteadas, para esto se observó la fila de “Se asumen varianzas iguales” o “No se asumen varianzas iguales”, ver Tabla 18-3, los resultados se observaron en la salida de la prueba de *Levene* para la igualdad de varianzas, estas se muestran inmediatamente:

Tabla 18-3: Prueba de Levene y Prueba t-Student en la consulta de datos

Prueba de muestras independientes										
		Prueba de Levene para la igualdad de varianzas		Prueba t para la igualdad de medias						
		F	Sig.	t	gl	Sig. (bilateral)	Diferencia de medias	Diferencia de error estándar	95% de intervalo de confianza de la diferencia	
									Inferior	Superior
Valor de Tiempo de Respuesta	Se asumen varianzas iguales	25,57	,000	8,43	18	,000	40,50	4,80	30,41	50,60

	No se asumen varianzas iguales			8,43	9,15	0,000	40,50	4,80	29,66	51,34
--	---	--	--	------	------	-------	-------	------	-------	-------

Realizado por: Fausto Orozco, 2017

Criterio para determinar Igualdad de Varianza:

Para determinar si las varianzas son iguales o no a través de las hipótesis planteadas se aplicaron los siguientes apartados:

$P\text{-Valor} > \alpha$ Aceptar H_0 = Las varianzas son iguales.

$P\text{-Valor} < \alpha$ Aceptar H_1 = Existe diferencia significativa entre las varianzas.

En seguida, se procedió a concluir si se cumple o no con el supuesto de igualdad de varianzas, como valor de probabilidad calculado se seleccionó el valor 0,000, que es el nivel de significancia de *Levene* (columna “Prueba de Levene para la igualdad de varianzas”), de la Tabla 18-3, para compararlo con el nivel si significancia $\alpha = 0,050$, ver Tabla 19-3:

Tabla 19-3: Decisión estadística de la igualdad de varianzas en la consulta de datos

IGUALDAD DE VARIANZA		
Valor de Probabilidad Obtenido	Relación	Nivel de Significancia
P-Valor = 0,000	<	$\alpha = 0,050$
CONCLUSIÓN:		
La varianza muestral sin el sistema y con el sistema tienen una diferencia estadísticamente significativa, es decir, se rechaza la hipótesis nula de igualdad de varianzas H_0 y se acepta la hipótesis alternativa H_1 .		

Realizado por: Fausto Orozco, 2017

Se concluyó entonces que las dos varianzas sin el sistema y con el sistema para la consulta de datos tenían una diferencia estadísticamente significativa, por esta razón, se rechazó la hipótesis nula H_0 y se aceptó la hipótesis alternativa H_1 . De tal forma que, se pudo realizar la prueba t de Student de comparación de dos medias, grupos o muestras independientes de carácter paramétrico.

5. Prueba t-Student para la igualdad de medias

Para el análisis de la comparación de medias de las dos variables cuantitativas tiempo, sin el sistema y con el sistema para la consulta de datos, se considerará la prueba de t-Student.

Después de comprobar la normalidad y la igualdad de varianza de las variables tiempo. Al finalizar el estudio estadístico, ya se puede dar una conclusión sobre si las medias del tiempo, con el sistema y sin el sistema para la consulta de datos, son iguales o existe una diferencia estadísticamente significativa entre las medias de las dos muestras.

Las hipótesis nula y alternativa para la igualdad de medias del tiempo, *sin* el sistema y con el sistema, para la consulta de datos quedaron formalmente enunciadas de la siguiente manera:

H_0 : Las medias del tiempo sin el sistema y con el sistema para la consulta de datos ***son iguales***.

H_1 : Las medias del tiempo sin el sistema y con el sistema para la consulta de datos ***no son iguales***.

Criterio para determinar la igualdad de medias:

Si la probabilidad obtenida P-Valor $> \alpha$, no rechace H_0 (Se acepta H_0).

Si la probabilidad obtenida P-Valor $< \alpha$, rechace H_0 (Se acepta H_1).

Por último, se procedió a concluir si se cumple con la hipótesis nula de igualdad de medias del tiempo H_0 de la prueba t-Student, como valor de probabilidad calculado se eligió el resultado 0,000, que es el valor del estadístico t de la distribución t-Student (columna “Prueba t para la igualdad de medias”), ver Tabla 18-3, y a su vez este valor será comparado con el valor de significancia $\alpha = 0,050$ previamente asignado, ver Tabla 20-3.

Tabla 20-3: Decisión estadística de la prueba t-Student para la igualdad de medias del tiempo en la consulta de datos

Prueba t-Student para la igualdad de medias del tiempo		
Valor de Probabilidad Obtenido	Relación	Nivel de Significancia
P-Valor = 0,000	<	$\alpha = 0,050$
CONCLUSIÓN: <i>Existe una diferencia estadísticamente significativa</i> entre el tiempo medio sin el sistema y el tiempo medio con el sistema para la consulta de datos, es decir, se rechaza la hipótesis nula de igualdad de medias H_0 y se acepta la hipótesis alternativa H_1 .		

Realizado por: Fausto Orozco, 2017

Según el análisis estadístico de la prueba t-Student se observó que existe una diferencia estadísticamente significativa entre la media del tiempo sin el sistema y la media del tiempo

con el sistema para la consulta de datos de las historias clínicas de los pacientes, por esta razón, se rechazó la hipótesis nula H_0 y se aceptó la hipótesis alternativa H_1 .

Haciendo uso de la herramienta de office, Excel 2016, los valores de la siguiente tabla se obtuvieron de aplicar *la prueba t de student para dos muestras suponiendo varianzas desiguales*.

Tabla 21-3: Datos estadísticos t-Student en la consulta de datos

Consulta de Datos		
Prueba t para dos muestras suponiendo varianzas desiguales		
	<i>Sin el sistema</i>	<i>Con el sistema</i>
Media	50,85	10,35
Varianza	228,825	1,89166667
Observaciones	10	10
Diferencia hipotética de las medias	0	
Grados de libertad	9	
<u>Estadístico t</u>	8,431707630	
<u>P(T<=t) una cola</u>	0,000007255	
<u>Valor crítico de t (una cola)</u>	1,833112933	
<u>P(T<=t) dos colas</u>	0,000014510	
<u>Valor crítico de t (dos colas)</u>	2,262157163	

Realizado por: Fausto Orozco, 2017

Tiempos:

Para un análisis comparativo en la mejora de los tiempos de respuesta con el empleo del sistema para el proceso de consulta de los datos de las historias clínicas de los pacientes, se aplicó el método matemático que permite obtener el factor de proporcionalidad del cuarto término desconocido a partir de los valores de los otros tres términos conocidos.

Esto es, se trata de una operación que nos ayuda a resolver rápidamente problemas de *proporcionalidad*, es decir, para este proceso necesitamos 3 datos: dos magnitudes proporcionales entre sí, la primera es el *tiempo medio total sin el sistema* con un valor de 50,85 segundos y la segunda es el *porcentaje del tiempo medio total sin el sistema* con un valor de 100 %, y una tercera magnitud que es el *tiempo medio total con el sistema* con un valor de 10,35 segundos. A partir de estos tres datos conocidos, averiguaremos el cuarto término desconocido de la proporcionalidad.

En consideración se puede observar el cálculo del cuarto término de la proporcionalidad, obteniendo de esta manera el nuevo *porcentaje del tiempo medio total con el sistema*, su resultado fue de 20,35 %.

$$\begin{array}{cc} 50,85 & 100 \% \\ 10,35 & X \end{array}$$

$$X = \frac{10,35 \times 100\%}{50,85} = 20,35 \%$$

Siendo el 100 % el porcentaje del tiempo medio total del proceso realizado manualmente sin el sistema, se procedió a realizar la diferencia con el 20,35 % que es el nuevo valor de porcentaje del tiempo medio total del proceso realizado automáticamente con el sistema, de donde se obtuvo el valor de 79,65 %, esto es:

$$100\% - 20,35\% = 79,65\%$$

Se concluyó por lo tanto que, en el proceso de consulta de los datos de las historias clínicas de los pacientes con el sistema, se ha mejorado el tiempo promedio en un valor del 79,65 %.

A continuación, se presenta un gráfico de barras en el que se observa el tiempo medio sin el sistema y el tiempo medio con el sistema, para la consulta de los datos de las historias clínicas de los pacientes.

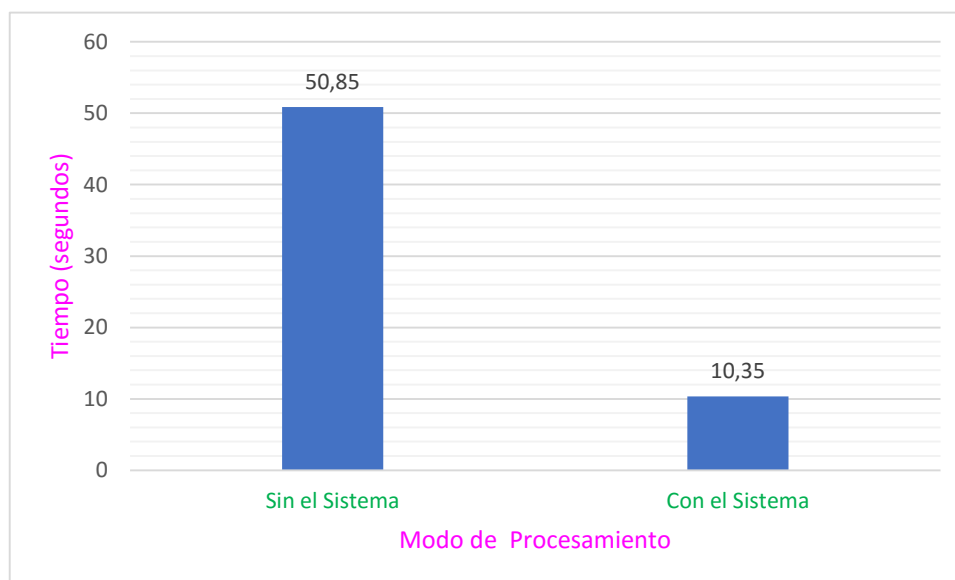


Gráfico 6-3 Tiempo de respuesta medio en la consulta de datos

Realizado por: Fausto Orozco, 2017

3.6 Análisis estadístico en la reserva de citas médicas

Posteriormente se procedió a obtener la muestra para realizar los análisis estadísticos correspondientes para el proceso de reserva de citas médicas de los pacientes, la muestra se consiguió de acuerdo al número de flujo de personas que llegaron al hospital para ser atendidos por un médico en un día de trabajo, esto es, se observó que en promedio 10 pacientes reservaron sus citas médicas con un determinado doctor para ser atendidos, por este motivo se decidió tomar una muestra con un valor igual a 10 para el análisis.

Tabla 22-3: Tiempo de respuesta en la reserva de citas médicas

Reserva Citas Médicas (segundos)		
Número de paciente	Sin el sistema	Con el sistema
1	70	14
2	45,6	12,3
3	78	13,8
4	36,1	11
5	62,6	14,7
6	48	13
7	56	11,5
8	42,8	12,5
9	34	15
10	61,8	13
Promedio	53,49	13,08

Realizado por: Fausto Orozco, 2017

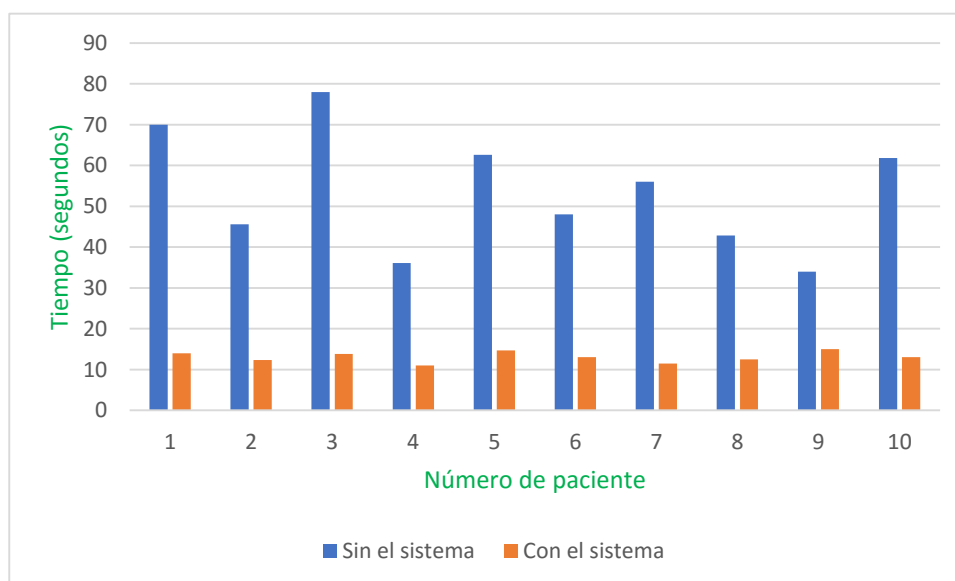


Gráfico 7-3 Tiempo de respuesta en la reserva de citas médicas

1. Redacción de la hipótesis

$H_1 = \textit{Existe}$ una diferencia significativa entre la media del tiempo sin el sistema para *reservar las citas médicas de los pacientes* y la media del tiempo con el sistema para *reservar las citas médicas de los pacientes*.

$H_0 = \textit{No Existe}$ una diferencia significativa entre la media del tiempo sin el sistema para *reservar las citas médicas de los pacientes* y la media del tiempo con el sistema para *reservar las citas médicas de los pacientes*.

2. Determinar el nivel significancia α

$$\alpha = 0,050$$

Es el porcentaje de error que se estuvo dispuesto a correr en la realización de la prueba estadística.

3. Elección de la prueba estadística

La prueba estadística que se utilizó es t-Student (muestras independientes) debido a que los datos de la muestra es 10 y considerando un análisis a dos colas.

4. Lectura de P-Valor

Para el proceso de reservación de citas médicas de los pacientes se realizó el mismo estudio para conocer si los datos que se obtuvieron siguieron una distribución normal, se hizo uso del software SPSS, el mismo que presentó los valores obtenidos para su respectivo análisis.

Para hacer uso de la distribución t-Student, fue fundamental que comprobemos, antes de empezar, si se daba el supuesto de normalidad para la reservación de una cita médica (esto es, si se ajusta a una distribución normal la variable tiempo para cada uno de los grupos).

NORMALIDAD

Para el estudio de la normalidad de las dos poblaciones, sin el sistema y con el sistema, se analizó la prueba de Shapiro Wilk ya que es ésta la que permitió trabajar con una muestra menor o igual a 30 observaciones, es decir, para la reservación de las citas médicas se consideró un valor de $n = 10$ observaciones. En la Tabla 23-3 se obtuvo las dos pruebas que permitieron comprobar si cumplen o no el supuesto de normalidad, esta fue obtenida desde el SPSS la cual exhibió sus resultados, y las pruebas presentadas son las siguientes:

- **Prueba de Kolmogorov-Smirnov** Permite trabajar con muestras grandes (>30 pacientes).
- **Prueba de Shapiro-Wilk** Permite trabajar con muestras pequeñas (<=30 pacientes).

Para ambas muestras tanto con el sistema y sin el sistema, se observó la salida de las pruebas de normalidad de *Shapiro-Wilk*, las mismas que se analizó:

Tabla 23-3: Prueba de normalidad de Shapiro-Wilk del tiempo de respuesta en la reservación de citas médicas

Pruebas de normalidad							
Tiempo de Respuesta		Kolmogorov-Smirnov			Shapiro-Wilk		
		Estadístico	gl	Sig.	Estadístico	gl	Sig.
Valor de Tiempo de Respuesta	Sin el Sistema	,146	10	,200*	,960	10	,791
	Con el Sistema	,124	10	,200*	,970	10	,889

Realizado por: Fausto Orozco, 2017

Criterio para determinar Normalidad:

Asumiendo como nivel de significación $\alpha = 0,05$, se consideraron las siguientes hipótesis estadísticas.

La hipótesis nula y alternativa para la variable cuantitativa, tiempo de respuesta para la reservación de una cita médica, *sin* el sistema, se representa a continuación:

H_0 : La variable aleatoria tiempo *sin* el sistema tiene una distribución normal.

H_1 : La variable aleatoria tiempo *sin* el sistema *no* tiene una distribución normal.

La hipótesis nula y alternativa para la variable cuantitativa, tiempo de respuesta para la reservación de una cita médica, *con* el sistema, se representa a continuación:

H_0 : La variable aleatoria tiempo *con* el sistema tiene una distribución normal.

H_1 : La variable aleatoria tiempo *con* el sistema *no* tiene una distribución normal.

Para las hipótesis planteadas se aplicó los siguientes apartados, para determinar si se acepta o se rechaza cada una estas:

$P\text{-Valor} > \alpha$ Aceptar $H_0 =$ Los datos provienen de una distribución **normal**.

$P\text{-Valor} < \alpha$ Aceptar $H_1 =$ Los datos **NO** provienen de una distribución **normal**.

Inmediatamente, se procedió a concluir si ambas muestras, tanto con el sistema y sin el sistema, cumplen con el supuesto de normalidad, como valor de probabilidad calculado se escogió el nivel de significancia de *Shapiro-Wilk*, ver Tabla 23-3, para el caso sin el sistema es 0,791 y lo comparamos con el nivel de significancia 0,050, y para el caso con el sistema es 0,889 igualmente lo comparamos con el nivel de significancia 0,050, ver Tabla 24-3.

Tabla 24-3: Decisión estadística de la normalidad de Shapiro-Wilk en la reservación de citas médicas

NORMALIDAD Tiempos		
Valor de Probabilidad Obtenido	Relación	Nivel de Significancia
P-Valor (<i>Sin el sistema</i>) = 0,791	>	$\alpha = 0,050$
P-Valor (<i>Con el sistema</i>) = 0,889	>	$\alpha = 0,050$
CONCLUSIÓN: La variable aleatoria tiempo en ambos grupos se comporta normalmente, es decir, se acepta la hipótesis nula de normalidad H_0 de la distribución en cada caso (sin el sistema y con el sistema).		

Realizado por: Fausto Orozco, 2017

Se concluyó que ambas distribuciones de datos, es decir, la muestra del tiempo sin el sistema y la muestra del tiempo con el sistema para la reservación de las citas médicas, proceden de una distribución normal, por esta razón, se acepta la hipótesis nula H_0 y se rechaza la hipótesis alternativa H_1 . De manera que, se procedió a realizar la prueba de Levene para verificar la igualdad de varianzas.

IGUALDAD DE VARIANZA

La prueba de Levene para la igualdad de varianzas nos indica si podemos o no suponer varianzas iguales, esto nos permitirá, conocer en qué caso nos encontramos, para hacer la lectura de los resultados en la fila “Se asumen varianzas iguales” o “No se asumen varianzas iguales” en la Tabla 25-3. De hecho, SPSS lo realiza automáticamente siempre que se desea realizar la prueba t-Student, y muestra los resultados para la prueba t-Student, tanto en el caso de que se desee admitir o rechazar la igualdad de varianzas.

Se partió de las hipótesis estadísticas para contrastar el supuesto de homocedasticidad u homogeneidad de la varianza, tomando como nivel de significación $\alpha = 0,05$.

Se describió, la hipótesis nula y alternativa para la igualdad de varianzas, *sin* el sistema y con el sistema, para la reservación de citas médicas:

H_0 : Las varianzas sin el sistema y con el sistema para la reservación de citas médicas **son iguales**.

H_1 : Las varianzas sin el sistema y con el sistema para la reservación de citas médicas **no son iguales**.

Se procedió a hacer un análisis en las dos hipótesis planteadas, para esto se observó la fila de “Se asumen varianzas iguales” o “No se asumen varianzas iguales”, ver Tabla 25-3, los resultados se observaron en la salida de la prueba de *Levene* para la igualdad de varianzas, estas se muestran inmediatamente:

Tabla 25-3: Prueba de Levene y Prueba t-Student en la reservación de citas médicas

Prueba de muestras independientes										
		Prueba de Levene para la igualdad de varianzas		Prueba t para la igualdad de medias						
		F	Sig.	t	gl	Sig. (bilateral)	Diferencia de medias	Diferencia de error estándar	95% de intervalo de confianza de la diferencia	
									Inferior	Superior
Valor de Tiempo de Respuesta	Se asumen varianzas iguales	25,48	,000	8,71	18	,000	40,41	4,64	30,67	50,15
	No se asumen varianzas iguales			8,71	9,15	,000	40,41	4,64	29,94	50,88

Realizado por: Fausto Orozco, 2017

Criterio para determinar Igualdad de Varianza:

Para determinar si las varianzas son iguales o no a través de las hipótesis planteadas se aplicaron los siguientes apartados:

$P\text{-Valor} > \alpha$ Aceptar H_0 = Las varianzas son iguales.

$P\text{-Valor} < \alpha$ Aceptar H_1 = Existe diferencia significativa entre las varianzas.

En seguida, se procedió a concluir si se cumple o no con el supuesto de igualdad de varianzas, como valor de probabilidad calculado se seleccionó el valor 0,000, que es el nivel de

significancia de *Levene* (columna “Prueba de Levene para la igualdad de varianzas”), de la Tabla 25-3, para compararlo con el nivel de significancia $\alpha = 0,050$, ver Tabla 26-3:

Tabla 26-3: Decisión estadística de la igualdad de varianzas en la reservación de citas médicas

IGUALDAD DE VARIANZA		
Valor de Probabilidad Obtenido	Relación	Nivel de Significancia
P-Valor = 0,000	<	$\alpha = 0,050$
CONCLUSIÓN: La varianza muestral sin el sistema y con el sistema tienen una diferencia estadísticamente significativa, es decir, se rechaza la hipótesis nula de igualdad de varianzas H_0 y se acepta la hipótesis alternativa H_1 .		

Realizado por: Fausto Orozco, 2017

Se concluyó entonces que las dos varianzas sin el sistema y con el sistema para la reservación de citas médicas tenían una diferencia estadísticamente significativa, por esta razón, se rechazó la hipótesis nula H_0 y se aceptó la hipótesis alternativa H_1 . De tal forma que, se pudo realizar la prueba t de Student de comparación de dos medias, grupos o muestras independientes de carácter paramétrico.

5. Prueba t-Student para la igualdad de medias

Para el análisis de la comparación de medias de las dos variables cuantitativas tiempo, sin el sistema y con el sistema para la reservación de citas médicas, se considerará la prueba de t-Student.

Después de comprobar la normalidad y la igualdad de varianza de las variables tiempo. Al finalizar el estudio estadístico, ya se puede dar una conclusión sobre si las medias del tiempo, con el sistema y sin el sistema para la reservación de citas médicas, son iguales o existe una diferencia estadísticamente significativa entre las medias de las dos muestras.

Las hipótesis nula y alternativa para la igualdad de las medias del tiempo, *sin* el sistema y con el sistema, para la reservación de citas médicas quedaron formalmente enunciadas de la siguiente manera:

H_0 : Las medias del tiempo sin el sistema y con el sistema para la reservación de citas médicas *son iguales*.

H_1 : Las medias del tiempo sin el sistema y con el sistema para la reservación de citas médicas *no son iguales*.

Criterio para determinar la igualdad de medias:

Si la probabilidad obtenida $P\text{-Valor} > \alpha$, no rechaza H_0 (Se acepta H_0).

Si la probabilidad obtenida $P\text{-Valor} < \alpha$, rechaza H_0 (Se acepta H_1).

Por último, se procedió a concluir si se cumple con la hipótesis nula de igualdad de medias del tiempo H_0 de la prueba t-Student, como valor de probabilidad calculado se eligió el resultado 0,000, que es el valor del estadístico t de la distribución t-Student (columna “Prueba t para la igualdad de medias”), ver Tabla 25-3, y a su vez este valor será comparado con el valor de significancia $\alpha = 0,050$ previamente asignado, ver Tabla 27-3.

Tabla 27-3: Decisión estadística de la prueba t-Student para la igualdad de medias del tiempo en la reservación de citas médicas

Prueba t-Student para la igualdad de medias del tiempo		
Valor de Probabilidad Obtenido	Relación	Nivel de Significancia
P-Valor = 0,000	<	$\alpha = 0,050$
CONCLUSIÓN: <i>Existe una diferencia estadísticamente significativa</i> entre el tiempo medio sin el sistema y el tiempo medio con el sistema para la reservación de citas médicas, es decir, se rechaza la hipótesis nula de igualdad de medias H_0 y se acepta la hipótesis alternativa H_1 .		

Realizado por: Fausto Orozco, 2017

Según el análisis estadístico de la prueba t-Student se observó que existe una diferencia estadísticamente significativa entre la media del tiempo sin el sistema y la media del tiempo con el sistema para la reservación de citas médicas de los pacientes, por esta razón, se rechazó la hipótesis nula H_0 y se aceptó la hipótesis alternativa H_1 .

Haciendo uso de la herramienta de office, Excel 2016, los valores de la siguiente tabla se obtuvieron de aplicar la prueba t de Student para dos muestras suponiendo varianzas desiguales.

Tabla 28-3: Datos estadísticos t-Student en la reserva de citas médicas

Reservación de Citas Médicas		
Prueba t para dos muestras suponiendo varianzas desiguales		
	Sin el sistema	Con el sistema
Media	53,49	13,08
Varianza	213,401	1,71733333
Observaciones	10	10

Diferencia hipotética de las medias	0	
Grados de libertad	9	
Estadístico t	8,712647394	
P(T<=t) una cola	0,000005562	
Valor crítico de t (una cola)	1,833112933	
P(T<=t) dos colas	0,000011125	
Valor crítico de t (dos colas)	2,262157163	

Realizado por: Fausto Orozco, 2017

Tiempos:

Para un análisis comparativo en la mejora de los tiempos de respuesta con el empleo del sistema para el proceso de consulta de los datos de las historias clínicas de los pacientes, se aplicó el método matemático que permite obtener el factor de proporcionalidad del cuarto término desconocido a partir de los valores de los otros tres términos conocidos.

Esto es, se trata de una operación que nos ayuda a resolver rápidamente problemas de *proporcionalidad*, es decir, para este proceso necesitamos 3 datos: dos magnitudes proporcionales entre sí, la primera es el *tiempo medio total sin el sistema* con un valor de 53,49 segundos y la segunda es el *porcentaje del tiempo medio total sin el sistema* con un valor de 100 %, y una tercera magnitud que es el *tiempo medio total con el sistema* con un valor de 13,08 segundos. A partir de estos tres datos conocidos, averiguaremos el cuarto término desconocido de la proporcionalidad.

En consideración se puede observar el cálculo del cuarto término de la proporcionalidad, obteniendo de esta manera el nuevo *porcentaje del tiempo medio total con el sistema*, su resultado fue de 24,45 %.

$$\begin{array}{cc} 53,49 & 100 \% \\ 13,08 & X \end{array}$$

$$X = \frac{13,08 \times 100\%}{53,49} = 24,45 \%$$

Siendo el 100 % el porcentaje del tiempo medio total del proceso realizado manualmente sin el sistema, se procedió a realizar la diferencia con el 24,45 % que es el nuevo valor de porcentaje del tiempo medio total del proceso realizado automáticamente con el sistema, de donde se obtuvo el valor de 75,55 %, esto es:

$$100\% - 24,45\% = 75,55\%$$

Se concluyó por lo tanto que, en el proceso de reservación de citas médicas de los pacientes con el sistema, se ha mejorado el tiempo promedio en un valor del 75,55 %.

A continuación, se presenta un gráfico de barras en el que se observa el tiempo medio sin el sistema y el tiempo medio con el sistema, para la reservación de citas médicas los pacientes.

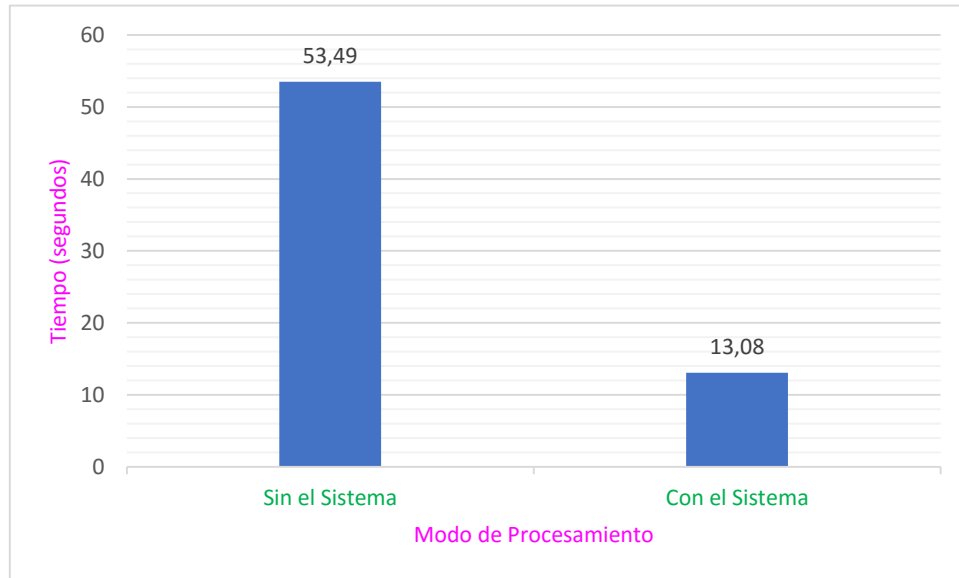


Gráfico 8-3 Tiempo de respuesta medio en la reservación de citas médicas
Realizado por: Fausto Orozco, 2017

3.7 Tiempos totales

Para finalizar con el estudio de los tiempos de respuesta en los procesos de inserción, actualización y consulta de las historias clínicas de los pacientes, así como en la reservación de las citas médicas de los pacientes, se procedió a calcular el tiempo medio total.

Tabla 29-3: Tiempo medio total

Operación	Sin el sistema Tiempo (segundos)	Con el sistema Tiempo (segundos)
Tiempo medio para la inserción de datos	202	47.5
Tiempo medio para la actualización de datos	128	28.1
Tiempo medio para la consulta de datos	50.85	10.35
Tiempo medio para la reservación de citas médicas	53.49	13.08
Tiempo medio total	434,34	99,03

Realizado por: Fausto Orozco, 2017

Para realizar un análisis comparativo en la mejora de los tiempos de respuesta con el empleo del sistema para los procesos de inserción, actualización y consulta de los datos de las

historias clínicas de los pacientes y además el proceso de la reservación de las citas médicas de los pacientes, se aplicó el método matemático que permite obtener el factor de proporcionalidad del cuarto término desconocido a partir de los valores de los otros tres términos conocidos.

Esto es, se trata de una operación que nos ayuda a resolver rápidamente problemas de *proporcionalidad*, es decir, para este proceso necesitamos 3 datos: dos magnitudes proporcionales entre sí, la primera es el *tiempo medio total sin el sistema* con un valor de 434,34 segundos y la segunda es el *porcentaje del tiempo medio total sin el sistema* con un valor de 100 %, y una tercera magnitud que es el *tiempo medio total con el sistema* con un valor de 99,03 segundos. A partir de estos tres datos conocidos, averiguaremos el cuarto término desconocido de la proporcionalidad.

En consideración se puede observar el cálculo del cuarto término de la proporcionalidad, obteniendo de esta manera el nuevo *porcentaje del tiempo medio total con el sistema*, su resultado fue de 22,80 %.

$$\begin{array}{cc} 434,34 & 100 \% \\ 99,03 & X \end{array}$$

$$X = \frac{99,03 \times 100\%}{434,34} = 22,80 \%$$

Siendo el 100 % el porcentaje del tiempo medio total del proceso realizado manualmente sin el sistema, se procedió a realizar la diferencia con el 22,80 % que es el nuevo valor de porcentaje del tiempo medio total del proceso realizado automáticamente con el sistema, de donde se obtuvo el valor de 77,20 %, esto es:

$$100\% - 22,80\% = 77,20\%$$

Se concluyó por lo tanto que, para los procesos de inserción, actualización y consulta de los datos de las historias clínicas de los pacientes y el proceso de la reservación de las citas médicas de los pacientes con el sistema, se ha mejorado el tiempo medio total en un valor del 77,20 %.

También, se presenta un gráfico de barras en el que se observa el tiempo medio total sin el sistema y el tiempo medio total con el sistema, para las operaciones de los datos de las historias clínicas y de la reservación de las citas médicas de los pacientes.

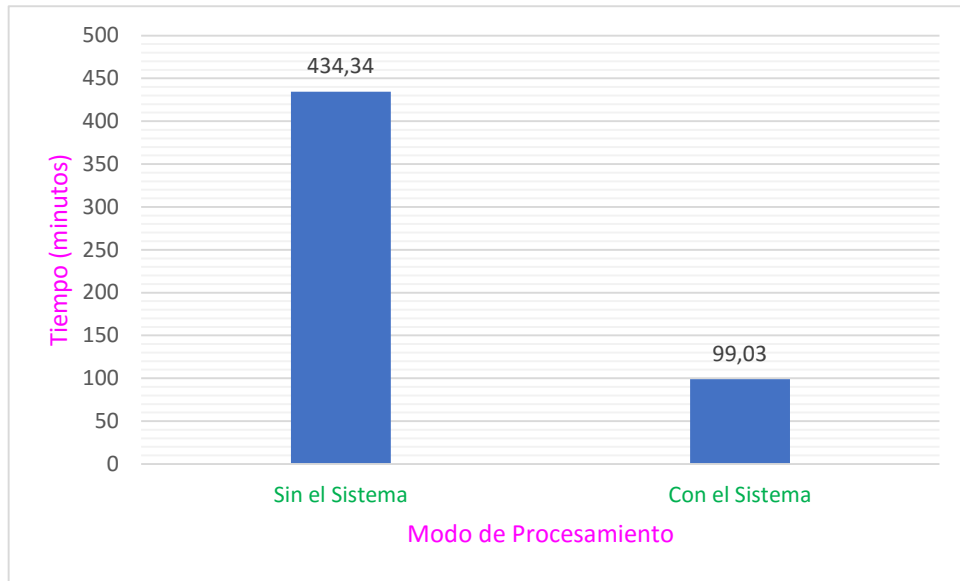


Gráfico 9-3 Tiempo de respuesta medio total

Realizado por: Fausto Orozco, 2017

CONCLUSIONES

- Analizando MariaDB se conoció que ofrece una alta compatibilidad con MySQL, así como el motor Aria que es el reemplazo de MyISAM, XtraDB que reemplaza al motor InnoDB y presentando mejoras de velocidad con el motor de almacenamiento Aria además de su distribución bajo licencia GLP v2 con la ventaja de ser open source, logrando así un bajo costo en la implementación del sistema utilizando este motor de base de datos.
- Se logró desarrollar el sistema informático obteniendo resultados positivos en el Hospital Básico San Marcos, con lo cual se cumple con el objetivo general, puesto que el sistema informático de gestión de historias clínicas implementado ha contribuido de manera eficiente en minimizar los tiempos de respuesta en las transacciones realizadas.
- El desarrollo del módulo que permite otorgar las citas médicas a los pacientes del hospital permitió la integración del DBMS MariaDB con la aplicación móvil Android, consiguiendo brindar un servicio de salud a través de un dispositivo móvil.
- Se concluye por lo tanto que, en el proceso de inserción de los datos de las historias clínicas de los pacientes con el sistema, se ha mejorado el tiempo promedio en un valor del 76,49 %.
- Se concluye por lo tanto que, en el proceso de actualización de los datos de las de las historias clínicas de los pacientes con el sistema, se ha mejorado el tiempo promedio en un valor del 78,05 %.
- Se concluye por lo tanto que, en el proceso de consulta de los datos de las historias clínicas de los pacientes con el sistema, se ha mejorado el tiempo promedio en un valor del 79,65 %.
- Se concluye por lo tanto que, en el proceso de reservación de citas médicas de los pacientes con la aplicación móvil, se ha mejorado el tiempo promedio en un valor del 75,55 %.
- Con la utilidad de la aplicación web se logró optimizar el tiempo medio total de 434,34 sg. que se demoraba inicialmente, a 99,03 sg. que se demora actualmente, llevando con esto a una mejora del tiempo medio total en un valor de 77,20 %.

RECOMENDACIONES

- Se debería revisar la documentación de MySQL ya que todos sus comandos, interfaces, librerías y APIs también existen en MariaDB por ser una ramificación de MySQL, con estos se permitirá crear, mantener, modificar, y manipular una base de datos.
- En el sistema de Gestión de Historias Clínicas del Hospital Básico San Marcos, se sugiere implementar otros módulos de importancia, como son: hospitalización, emergencias, resultados de laboratorio y más, que permitan mejorar los servicios de la institución.
- En la aplicación móvil Android, se debería incrementar los servicios de salud, esto permitiría que el paciente pueda beneficiarse de servicios en línea cuando lo necesite, como es el disponer de una receta médica, historia clínica, y otros.
- Es preciso investigar sobre las ventajas, desventajas y si existe o no información de las posibles herramientas a utilizar para el desarrollo de software, ya que esto permitirá dar una alternativa de solución a los requerimientos del sistema.
- Aplicar correctamente la metodología Scrum para evitar demoras en el desarrollo del proyecto. Siendo el análisis de requerimientos muy importante para la ejecución e implementación de los mismos en su desarrollo.

GLOSARIO

API: Una API es un conjunto de funciones y procedimientos que cumplen una o muchas funciones con el fin de ser utilizadas por otro *software*. Las siglas API vienen del inglés *Application Programming Interface*. En español sería Interfaz de Programación de Aplicaciones. Una API nos permite implementar las funciones y procedimientos que engloba en nuestro proyecto sin la necesidad de programarlas de nuevo. En términos de programación, es una capa de abstracción. (hipertextual,2014)

CAISE Centro de Atención Integral en Salud de la Escuela Superior Politécnica de Chimborazo.

Código abierto: Plataforma de desarrollo de software libre.

DBMS: (Data Base Management System). **SGBD** (Sistema de Gestión de Base de Datos) Software que controla la organización, almacenamiento, recuperación, seguridad e integridad de los datos en una base de datos. Acepta solicitudes de la aplicación y ordena al sistema operativo transferir los datos apropiados (mastermagazine, 2010).

EJB: (Enterprise JavaBeans). Una aplicación EJB puede ser desplegada en cualquier servidor de aplicaciones que soporte J2EE. Reusabilidad de componentes. Una aplicación EJB está formada por componentes enterprise beans (bloque de construcción reusable). (Universidad de Alicante, 2004)

FORK: Una bifurcación, en el ámbito del desarrollo de software, es la creación de un proyecto en una dirección distinta de la principal u oficial tomando el código fuente del proyecto ya existente. (Borja, 2015)

GPL: (Licencia Pública General) o más conocida por su nombre en inglés GNU General Public License Versión 2, pretende garantizar su libertad para compartir y cambiar el software libre para asegurarse de que el software es gratuito para todos sus usuarios. Esta Licencia Pública General se aplica a la mayor parte del software de la Free Software Foundation y a cualquier otro programa cuyos autores se comprometan a usarlo. (Open Source Initiative, 1991)

HTML: HyperText Markup Language (lenguaje de marcas de hipertexto), HTML es el lenguaje de marcado estándar utilizado para crear páginas web y sus elementos forman los bloques de construcción de todos los sitios web. (W3C, 2016)

Java EE: Es una plataforma de desarrollo para el lenguaje de programación Java. (Groussard, 2010).

JDBC: La API de JDBC (Java Database Connectivity) es el estándar de la industria para la conectividad independiente de la base de datos entre el lenguaje de programación Java y una amplia gama de bases de datos. Bases de datos SQL y otras fuentes de datos tabulares, como hojas de cálculo o archivos planos. La API JDBC proporciona una API de nivel de llamada para el acceso a bases de datos basadas en SQL. (Oracle).

JMS: La API Java Message Service (JMS) es un estándar de mensajería que permite que los componentes de aplicación basados en Java Enterprise Edition (Java EE) creen, envíen, reciban y lean mensajes. Permite una comunicación distribuida que está ligeramente acoplada, confiable y asíncrona. (Oracle)

JPA: Java Persistence Api es una parte de la especificación de EJB 3. (Álvarez, 2014)

JSF: JavaServer Faces es el framework oficial de Java Enterprise para el desarrollo de interfaces de usuario avanzadas en aplicaciones web. (Departamento de ciencias de la Computación e IA, 2014).

Kernel: Se refiere al núcleo o centro sobre el cual está construido GlassFish.

NDK: El Android NDK (Native Development Kit) permite a los desarrolladores reutilizar código escrito en C/C++ introduciéndolo en las aplicaciones a través de JNI (Java Native Interface). El NDK hace que la ejecución de la aplicación sea en cierto modo más rápida, ya que pasará a ejecutarse directamente en el procesador y no es interpretado por una máquina virtual. (Pérez Esteso, 2014)

Oracle: es básicamente una herramienta cliente/servidor para la gestión de Bases de Datos. (Masip, 2002)

PostgreSQL: es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. (Martínez, 2010)

RMI: El sistema Java Remote Method Invocation (RMI) permite a un objeto que se ejecuta en una máquina virtual Java invocar métodos en un objeto que se ejecuta en otra máquina virtual Java. RMI proporciona comunicación remota entre los programas escritos en el lenguaje de programación Java. (Oracle)

SDK: SDK responde a las siglas Software Development Kit, lo que viene a ser un kit de desarrollo de software. Con él podremos desarrollar aplicaciones y ejecutar un emulador del sistema Android de la versión que sea. Todas las aplicaciones Android se desarrollan en lenguaje Java con este kit. Con SDK de Android podremos utilizar nuestro dispositivo desde nuestro ordenador. (Valencia, 2012)

Servlets: Los Servlets son módulos escritos en Java que se utilizan en un servidor, que puede ser o no ser servidor web, para extender sus capacidades de respuesta a los clientes al utilizar las potencialidades de Java. Los Servlets son para los servidores lo que los applets para los navegadores, aunque los servlets no tienen una interfaz gráfica. (Barrios, 2001)

SO: Los sistemas operativos para dispositivos móviles suelen ser menos robustos que los diseñados para las computadoras de escritorio o portátiles. Es decir, que con un dispositivo móvil no puedes hacer todo lo que haces con un computador o un portátil. (GCF AprendeLibre, 2014)

SQL Server: SQL Server es un sistema de gestión de bases de datos relacionales (RDBMS) de Microsoft que está diseñado para el entorno empresarial. SQL Server se ejecuta en T-SQL (Transact-SQL), un conjunto de extensiones de programación de Sybase y Microsoft que añaden varias características a SQL estándar, incluyendo control de transacciones, excepción y manejo de errores, procesamiento fila, así como variables declaradas. (TechTarget, 2015)

BIBLIOGRAFÍA

ABLESON, Frank. *Introducción al desarrollo en Android* [en línea]. 2013. [Consulta: 3 de mayo 2016]. Disponible en: <https://www.ibm.com/developerworks/ssa/library/os-android-devel/>

ÁLVAREZ, Cecilio. *JPA vs Hibernate*. [en línea]. 2014. [Consulta: 18 de marzo 2016]. Disponible en: <https://www.genbetadev.com/frameworks/jpa-vs-hibernate>

BAHIT, Eugenia. *Introducción al desarrollo ágil con Scrum* [en línea]. 2011. [Consulta: 22 de febrero 2016]. Disponible en: <http://www.desarrolloweb.com/articulos/desarrollo-agil-scrum.html>

BARRIOS, Juan Manuel. *Java Servlets*. [en línea]. 2001. [Consulta: 19 de marzo 2016]. Disponible en: <https://users.dcc.uchile.cl/~jbarrios/servlets/general.html>

BORJA, RIKARDO JOAS. *Bifurcación (desarrollo de software)*. [en línea]. 2015. [Consulta: 19 de marzo 2016]. Disponible en: <https://prezi.com/k31gpazaqzcf/bifurcacion-borja-ricardo/>

CCIA, D. *JSP Básico*. [en línea]. 2005. [Consulta: 9 de marzo 2016]. Disponible en: <http://www.jtech.ua.es/j2ee/2006-2007/doc/sesion08-apuntes.pdf>

CONOCIMIENTOSWEB.NET. *Características generales de los dispositivos móviles* [en línea]. 2015. [Consulta: 3 de mayo 2016]. Disponible en: <http://www.conocimientosweb.net/dcmt/ficha25959.html>

CHÁVEZ, Carlos.; & ALCANTARA, Cristhian. *MariaDB*. [blog]. 2015. [Consulta: 10 de enero 2016]. Disponible en: <http://conociendomariadb.blogspot.com>

DANS, Enrique. *MariaDB, el software libre y el lucro cesante*. [blog]. 2013. [Consulta: 15 de enero 2016]. Disponible en: <http://www.enriquedans.com/2013/05/mariadb-el-software-libre-y-el-lucro-cesante.html>

DATAPRIX. *Conexión y uso de base de datos en lenguaje Java*. [en línea]. 2006. [Consulta: 23 de febrero 2016]. Disponible en: <http://www.dataprix.com/2-conexion-uso-bases-datos-lenguaje-java>

DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN E IA. *Introducción a Java Server Faces*. [en línea]. 2014. [Consulta: 18 de marzo 2016]. Disponible en: <http://www.jtech.ua.es/j2ee/publico/jsf-2012-13/sesion01-apuntes.html>

DEVELOPERS. *AsyncTask* [en línea]. 2011. [Consulta: 2 de mayo 2016]. Disponible en: <https://developer.android.com/reference/android/os/AsyncTask.html>

ESTEBAN, Ángel. *TECNOLOGÍAS DE SERVIDOR CON JAVA: SERVLETS, JAVABEANS, JSP*. Madrid-España: Grupo EIDOS, 2000, pp.13-21.

GCF AprendeLibre. *Sistemas operativos para dispositivos móviles* [en línea]. 2014. [Consulta: 25 de febrero 2016]. Disponible en: https://www.gcfaprendelibre.org/tecnologia/curso/informatica_basica/sistemas_operativos/5.do

GIRONÉS, J. *El gran libro de Android*. Segunda Edición. Barcelona-España: Marcombo, S.A. 2012, pp. 61-91.

GITHUB. *What is WPN-XM*. [en línea]. 2015. [Consulta: 25 de febrero 2016]. Disponible en: <https://github.com/WPN-XM/WPN-XM>

GLASSFISH. [en línea]. 2015. [Consulta: 11 de enero 2016]. Disponible en: <http://www.ecured.cu/GlassFish>

GONZÁLEZ ÁVALOS, Susana. *Usos y aplicaciones de bases de datos*. [blog]. 2011. [Consulta: 12 de enero 2016]. Disponible en: <https://susanagonzalezavalos.wordpress.com/070211-usos-y-aplicaciones-de-base-de-datos/>

GROUSSARD, Thierry. *Java Enterprise Edition: Desarrollo de aplicaciones web con JEE 6*. Barcelona-España: Ediciones ENI, 2010, p15.

HIPERTEXTUAL. *¿Qué es una API?* [en línea]. 2014. [Consulta: 5 de mayo 2016]. Disponible en: <https://hipertextual.com/archivo/2014/05/que-es-api/>

INFORMATICA. *LENGUAJE ASP*. [blog]. 2008. [Consulta: 4 de mayo 2016]. Disponible en: <http://aspanaypaty.blogspot.com/2008/12/lenguaje-asp.html>

INVARATO, Ramón. *AsyncTask en Android*. [en línea]. 2013. [Consulta: 2 de mayo 2016]. Disponible en: <https://jarroba.com/async-task-en-android/>

JAVA. *¿Qué es la tecnología Java y para qué la necesito.?* [en línea]. 2009. [Consulta: 15 de enero 2016]. Disponible en: https://www.java.com/es/download/faq/whatis_java.xml

LANCETALENT. *Los 3 tipos de aplicaciones móviles: ventajas e inconvenientes*. [blog]. 2014. [Consulta: 5 de mayo 2016]. Disponible en: <https://www.lancetalent.com/blog/tipos-de-aplicaciones-moviles-ventajas-inconvenientes/>

MARIADB. *About MariaDB*. [en línea]. 2012. [Consulta: 1 de marzo 2016]. Disponible en: <https://mariadb.com/kb/en/mariadb/about-mariadb/>

MARIADB. *MariaDB versus MySQL - Características*. [en línea]. 2011. [Consulta: 1 de mayo 2016]. Disponible en: <https://mariadb.com/kb/es/mariadb-versus-mysql-features/>

MARIADB. *MariaDB versus MySQL – Features*. [en línea]. 2014. [Consulta: 1 de marzo 2016]. Disponible en: <https://mariadb.com/kb/en/mariadb/mariadb-vs-mysql-features/>

MARTÍNEZ, Rafael. *Sobre PostgreSQL*. [en línea]. 2010. [Consulta: 19 de Marzo 2016]. Disponible en: http://www.postgresql.org.es/sobre_postgresql

MASIP, David. *Qué es Oracle*. [en línea]. 2002. [Consulta: 19 de marzo 2016]. Disponible en: <http://www.desarrolloweb.com/articulos/840.php>

MASTERMAGAZINE. *Definición de DBMS*. [en línea]. 2010. [Consulta: 17 de enero 2016]. Disponible en: <http://www.mastermagazine.info/termino/4544.php>

MERINO, María.; PÉREZ PORTO, Julián. *Definición de tratamiento*. [en línea]. 2013. [Consulta: 3 de mayo 2016]. Disponible en: <http://definicion.de/tratamiento/>

MINISTERIO DE SALUD PÚBLICA. *Manual del manejo, archivo de las historias clínicas* [en línea]. 2007. [Consulta: 5 de mayo 2016]. Disponible en: http://instituciones.msp.gob.ec/somossalud/Documentos_Financiero/2.1.%20Manual%20manejo%20historia%20clinica.pdf

MINISTERIO DE SANIDAD, SERVICIOS SOCIALES E IGUALDAD. *Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud, Décima Revisión*”, vols.1,2 y 3. [en línea]. España, 2010. [Consulta: 5 de mayo 2016]. Disponible en: http://eciemaps.mspsi.es/ecieMaps/browser/index_10_2008.html

OPEN SOURCE INITIATIVE. *GNU General Public License Version 2.* [en línea]. 1991. [Consulta: 20 de marzo 2016]. Disponible en: <https://opensource.org/licenses/gpl-2.0.php>

ORACLE. *The Java Database Connectivity (JDBC).* [en línea]. [Consulta: 18 de marzo 2016]. Disponible en: <http://www.oracle.com/technetwork/java/javase/jdbc/index.html>

ORACLE. *Java Message Service (JMS).* [en línea]. [Consulta: 18 de marzo 2016]. Disponible en: <http://www.oracle.com/technetwork/java/jms/index.html>

ORACLE. *Trail: RMI.* [en línea]. [Consulta: 19 de marzo 2016]. Disponible en: <http://docs.oracle.com/javase/tutorial/rmi/index.html>

PALACIOS AGUINAGA, Jennifer R. *Motivo de consulta* [en línea]. 2013. [Consulta: 4 de mayo 2016]. Disponible en: <http://es.slideshare.net/yukijenn/motivo-de-consulta>

PAVÓN MESTRAS, Juan. *Java EE Aplicaciones Web/Sistemas Web* [en línea]. 2013. [Consulta: 4 de mayo 2016]. Disponible en: <https://www.fdi.ucm.es/profesor/jpavon/web/41-J2EE.pdf>

PÉREZ, José Luis. *Principales lenguajes de programación web, ventajas y desventajas.* [en línea]. 2015. [Consulta: 5 de mayo 2016]. Disponible en: <http://www.registrodominiosinternet.es/2013/08/lenguajes-programacion-web-ventajas.html#more>

PÉREZ ESTESO, Mario. *Qué es el Android NDK [Parte 1].* [en línea]. 2014. [Consulta: 3 de mayo 2016]. Disponible en: <https://geekytheory.com/que-es-el-android-ndk-parte-1>

PRESSMAN, Roger S. *Ingeniería del Software Un Enfoque Práctico Quinta Edición* [en línea]. 2008. [Consulta: 10 de mayo 2016]. Disponible en: http://aleloj.weebly.com/uploads/9/3/6/4/936494/roger_pressman-ingeniera_del_software-v_ed-cap1.pdf

PRIETO BLÁZQUEZ, Josep.; et al., *Introducción a los dispositivos móviles*. [en línea]. 2011. [Consulta: 4 mayo 2016]. Disponible en: <http://openaccess.uoc.edu/webapps/o2/handle/10609/9582>

RAJKUMAR. *Agile Scrum Methodology In Software Development*. [en línea]. 2016. [Consulta: 10 de mayo 2016]. Disponible en: <http://www.softwaretestingmaterial.com/agile-scrum-methodology/>

RIBEIRO, R. *MariaDB Beginners Guide*. 2014.

RUBIO HURTADO, María José.; et al., *Cómo aplicar las pruebas paramétricas bivariadas t de Student y ANOVA en SPSS. Caso práctico*. [en línea]. 2012. [Consulta: 1 de marzo 2017]. Disponible en: www.raco.cat/index.php/REIRE/article/download/255792/342835

RUIZ URRACA, David. *Desarrollo de una aplicación móvil cliente-servidor basada en Android para la configuración y control de parámetros de un vehículo particular* [en línea]. 2013. [Consulta: 3 de mayo 2016]. Disponible en: http://biblioteca.unirioja.es/tfe_e/R000001713.pdf

RUSSELL, J.T. D. *Learning MySQL and MariaDB*. Estados Unidos: O'Reilly, 2015, pp.3-7.

SCHWABER, K., & SUTHERLAND, J. *La Guía Definitiva de Scrum: Las Reglas del Juego*. Creative Commons. 2013.

SERRA MANCHADO, David. *Estudio del servidor de aplicaciones Glassfish y de las aplicaciones J2EE*. [en línea]. 2010. [Consulta: 11 de enero 2016]. Disponible en: <https://core.ac.uk/download/pdf/13325903.pdf>

SOFTENG *Proceso y roles de Scrum*. [en línea]. 2010. [Consulta: 4 de marzo 2016]. Disponible en: <https://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum/proceso-roles-de-scrum.html>

SOSA SOSA, Víctor Jesús *Java Server Pages (JSP)* [en línea]. 2014. [Consulta: 4 de mayo 2016]. Disponible en: http://www.tamps.cinvestav.mx/~vjsosa/clases/sd/DAAI_JSP.pdf

TACURI UQUILLAS, A., & GRANIZO RODRÍGUEZ, V. [en línea]. Escuela Superior Politécnica de Chimborazo, Facultad de Salud Pública, Escuela de Medicina, Riobamba-Ecuador, 2005. [Consulta: 11 de enero 2016]. Disponible en: <http://medicina.esPOCH.edu.ec/>

TECHTARGET. *SQL Server*. [en línea]. 2015. [Consulta: 5 de mayo 2016]. Disponible en: <http://searchdatacenter.techtarget.com/es/definicion/SQL-Server>

UNIVERSIDAD DE ALICANTE, *Introducción a la tecnología EJB*. [en línea]. 2004. [Consulta: 20 de marzo 2016]. Disponible en: <http://www.jtech.ua.es/j2ee/2003-2004/abierto-j2ee-2003-2004/ejb/sesion01-apuntes.htm>

UNIVERSIDAD DE CÓRDOBA, *Enfermedad Información del término*. [en línea]. 2015. [Consulta: 4 de mayo 2016]. Disponible en: <http://www.uco.es/servicios/dgppa/images/prevencion/glosarioprl/fichas/e/Enfermedad.html>

UNIVERSIDAD POLITÉCNICA DE VALENCIA. *Proceso de desarrollo de software*. [en línea]. 2002. [Consulta: 10 de mayo 2016]. Disponible en: www.dsic.upv.es/asignaturas/facultad/lsi/doc/IntroduccionProcesoSW.doc

VALENCIA, Isabel. *Android para Principiantes ¿Qué quiere decir SDK de Android?*. [en línea]. 2012. [Consulta: 19 de marzo 2016]. Disponible en: <http://www.androidpit.es/sdk-android>

VOZIDEA. *Qué es MariaDB y ventajas frente a MySQL*. [en línea]. 2013. [Consulta: 19 de enero 2016]. Disponible en: <http://www.vozidea.com/que-es-mariadb-y-ventajas-frente-mysql>

W3C. *What is HTML?*. [en línea]. 2016. [Consulta: 18 de marzo 2016]. Disponible en: <https://www.w3.org/html/>

WPN-XM. *What is WPN-XM?*. [en línea]. 2015. [Consulta: 25 de febrero 2016]. Disponible en: <http://wpn-xm.org/>